PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-052572
(43)Date of publication of application : 24.02.1998

(51)Int.Cl. A63F 9/22

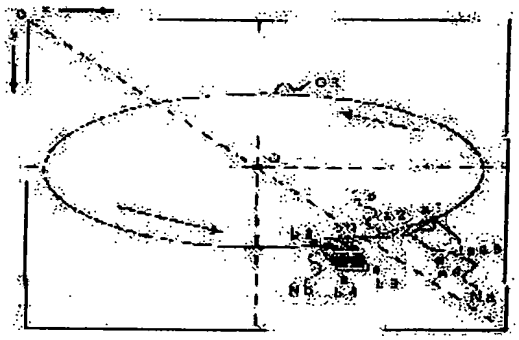
(21)Application number : 09-147052
(22)Date of filing : 22.05.1997

(71)Applicant : KONAMI CO LTD
(72)Inventor : OSHIMA NAOTO

(30)Priority
Priority number : 08149714 Priority date : 22.05.1996 Priority country : JP

(54) THROW GUIDE DISPLAY METHOD, GAME SYSTEM AND RECORDING MEDIUM FOR THROWING GAME

(57)Abstract:
PROBLEM TO BE SOLVED: To let a game player recognize a direction to throw an object to be thrown by displaying a throw guide image showing the throwing direction of the object to be thrown to be successively changed with the operation of a throwing object for throwing the object to be thrown.
SOLUTION: This method comprises guiding the timing for the throwing object in a game space to throw the object to be thrown in the throwing direction intended by the game player easily to comprehend for the game player. This purpose is achieved by displaying an arrow Na showing the direction for the throwing object to throw the object to be thrown and an orbit OR of this arrow Na on the display screen of a television monitor as shown in a figure A. Further, the direction shown by the arrow Na is coincident with the throwing direction of the object to be throws showing the attitude of a player, etc., to be displayed together with that arrow Na. Namely, the arrow Na showing the throwing direction corresponding to the throwing object to be displayed on the display screen with the passage of time is displayed together with the throwing object with the passage of time.



LEGAL STATUS
[Date of request for examination] 18.12.1998
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number] 3145058

[Date of registration] 05.01.2001

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]
[Claim 1] The image information which shows a throw object, and the image in which the throw-ed object by which a throw is visually carried out with the throw object concerned is shown are displayed on the screen of a display means at least. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object visually, while being operated based on actuation of an actuation means Based on actuation of the above-mentioned actuation means, the above-mentioned throw-ed object is used in the throw game by which a throw is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means. The throw guide method of presentation in the throw game which displays the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which is the throw guide method of presentation in a throw game, and changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown.
[Claim 2] Actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object is the throw guide method of presentation in the throw game according to claim 1 which is the actuation or abbreviation rotation actuation describing an arc.

[Claim 3] The above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 2 by which it is indicated by sequential so that the orbit according to actuation of the above-mentioned throw object may be drawn.
[Claim 4] The throw guide method of presentation in the throw game according to claim 2 by which it is indicated by sequential so that the locus image in which the locus of the above-mentioned throw guide image is shown may draw the orbit according to actuation of the above-mentioned throw object.
[Claim 5] The above-mentioned locus image is the throw guide method of presentation in the throw game according to claim 4 generated based on the coordinate information on a current throw guide image, and the coordinate information on the throw guide image displayed before predetermined time.
[Claim 6] The above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 1 by which a false three-dimensional display is carried out on the screen of the above-mentioned display means.
[Claim 7] The configuration of the above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 1 which is the configuration which is similar to an arrow-head configuration or this at least, and which has the visual direction directions function.
[Claim 8] The throw guide method of presentation in the throw game according to claim 1 to which adjustable [of the color of the above-mentioned throw guide image] is carried out according to rotation of the above-mentioned throw object at least.

[Claim 9] The color of the above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 8 which changes from cold color to warm color according to advance of rotation of the above-mentioned throw object.
[Claim 10] The image information which shows a throw object, and the image in which the throw-ed object by which a throw is visually carried out with the throw object concerned is shown are displayed on the screen of a display means at least. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object visually, while being operated based on actuation of an actuation means Based on actuation of the above-mentioned actuation means, the above-mentioned throw-ed object is used in the throw game by which a throw is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means. The throw guide method of presentation in the throw game containing the direction display step of a throw which displays the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which is the throw guide method of presentation in a throw game, and changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown.
[Claim 11] The migration and the rotation acquisition step which calculates the movement magnitude and the rotation of the above-mentioned throw guide image based on the throw include-angle information on the above-mentioned throw-ed object based on actuation of the above-mentioned actuation means in the above-mentioned direction display step of a throw, The address-data acquisition step which is based on the coordinate information on the above-mentioned throw guide, and the movement magnitude and rotation information which were searched for in the above-mentioned migration and rotation acquisition step, and acquires the address data on the memory of the above-mentioned throw guide, At least The address data on the memory of the representation point of the above-mentioned throw guide image, The throw guide method of presentation in the throw game containing the drawing directions step which directs drawing by supplying the information for specifying the color of the above-mentioned throw guide, and the texture information on the above-mentioned throw guide to a drawing processing means according to claim 10.
[Claim 12] The throw guide method of presentation in the throw game according to claim 11 to which adjustable [of the color of the above-mentioned throw guide image] is carried out according to rotation of the above-mentioned throw object at least.
[Claim 13] The color of the above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 12 by which sequential assignment is carried out to changing from cold color to warm color according to advance of rotation of the above-mentioned throw object.
[Claim 14] The locus generation step which generates the locus image in which the locus of the above-mentioned throw guide is shown is prepared further. The locus generation step concerned The storage step which memorizes the address data on the memory of a throw guide image, The address data for one extracted from the address data on the memory of current and the past two throw guide images at least, The throw guide method of presentation in the throw game containing the locus image drawing directions step which directs drawing of the above-mentioned locus image by supplying the information for specifying a color, and the texture information on the above-mentioned guide image to a drawing processing means according to claim 10.
[Claim 15] Actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object is the throw guide method of presentation in the throw game according to claim 10 which is the actuation or abbreviation rotation actuation describing an arc.
[Claim 16] The above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 15 by which it is indicated by sequential so that the orbit

according to actuation of the above-mentioned throw object may be drawn.
[Claim 17] The throw guide method of presentation in the throw game according to claim 15 by which it is indicated by sequential so that the locus image in which the locus of the above-mentioned throw guide image is shown may draw the orbit according to actuation of the above-mentioned throw object.
[Claim 18] The above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 10 by which a false three-dimensional display is carried out on the screen of the above-mentioned display means.
[Claim 19] The configuration of the above-mentioned throw guide image is the throw guide method of presentation in the throw game according to claim 10 which is the configuration which has the visual direction directions function which is similar to an arrow-head configuration or this at least.
[Claim 20] A display means to display at least the image information which shows a throw object, and the image information which shows the throw-ed object by which a throw is visually carried out with the throw object concerned, Actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object on the screen of the above-mentioned display means based on actuation of an actuation means and the above-mentioned actuation means, It is a game system equipped with the control means which performs the throw of the above-mentioned throw-ed object in a list visually. The above-mentioned control means The game system which has a direction display means of a throw to display the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown.
[Claim 21] Actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object is a game system according to claim 20 which is the actuation or abbreviation rotation actuation describing an arc.
[Claim 22] The above-mentioned throw guide image is a game system according to claim 21 by which it is indicated by sequential so that the orbit according to actuation of the above-mentioned throw object may be drawn.
[Claim 23] The game system according to claim 21 by which it is indicated by sequential so that the locus image in which the locus of the above-mentioned throw guide image is shown may draw the orbit according to actuation of the above-mentioned throw object.
[Claim 24] The above-mentioned locus image is a game system according to claim 23 generated based on the coordinate information on a current throw guide image, and the coordinate information on the throw guide image displayed before predetermined time.
[Claim 25] The above-mentioned throw guide image is a game system according to claim 20 by which a false three-dimensional display is carried out on the screen of the above-mentioned display means.
[Claim 26] The configuration of the above-mentioned throw guide image is a game system according to claim 20 which is the configuration which is similar to an arrow-head configuration or this at least, and which has the visual direction directions function.
[Claim 27] The game system according to claim 20 by which adjustable [of the color of the above-mentioned throw guide image] is carried out according to rotation of the above-mentioned throw object at least.
[Claim 28] The color of the above-mentioned throw guide image is a game system according to claim 27 changed from cold color to warm color according to advance of rotation of the above-mentioned throw object.
[Claim 29] The image information which shows a throw object, and the image in which the throw-ed object by which a throw is visually carried out with the throw object concerned is shown are displayed on the screen of a display means at least. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object

visually, while being operated based on actuation of an actuation means It is the record medium with which the throw game data which include the game program to which the throw of the above-mentioned throw-ed object is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means bas

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention] This invention is applied to the game system using an optical disk, a magnetic disk, a cassette type record medium using semiconductor memory, etc. with which game data were recorded, and relates to the throw guide method of presentation in a suitable throw game, a game system, and a record medium.
[0002]
[Description of the Prior Art] Many game systems are proposed. It is the system which consists of the system which consists of a special-purpose machine and a television monitor for home use, a business-use special-purpose machine, a personal computer or a workstation, a display, and a voice output machine. A controller for a player to operate each of these systems, The record medium with which the game data which consist of game program data and data, such as an image and voice, were recorded, It consists of CPU which performs control for generation of voice or an image based on game program data, the processor for processing an image, a processor for processing voice, CRT for displaying an image, and a loudspeaker for outputting voice. As the above-mentioned record medium, there are many cassettes which contained CD-ROM, semiconductor memory, and semiconductor memory. The game structure of a system is as above.
[0003] on the other hand -- the class of game -- a way of an increment -- following -- moreover, the contents of the game -- day by day -- complexity -- and it is diversified. Recently, it is proposed by operating a controller and moving the player on a screen to what performs a sport in false on the game space formed on the screen of a television monitor. As a sport game, team event games, such as soccer and baseball, and an individual event game are proposed. As an individual event game, when it divides roughly, there are "a game which runs", "a game which swims", "a game to which the player itself flies", "a game which lifts an object", "a game which fights", "a game which applies an object on a target", and "a game whose player itself throws an object (throw)." In addition, as a throwing event, the "shot put", the "hammer throw", the "discus throw", the "javelin throw", etc. occur. When these individual events are realized as a TV game, the gestalt can predict that it is as follows. That is, the control section of a game system makes the player in the game space formed on the screen of a television monitor etc. contest by moving visually based on the contents of actuation, and an actuation condition, when a game player operates a controller.
[0004]
[Problem(s) to be Solved by the Invention] Among the above-mentioned throwing events, the shot put is performed by throwing a ball, after a player rotates 180 abbreviation. The discus throw and the hammer throw are performed by throwing a disk and a hammer, after a player does multiple-times rotation into game area. In order to realize such a throwing event as a game In the processing which sets up a player's throw energy by a certain actuation of a

controller at least, and a screen top A player's image with half-rotation (shot put) or the processing displayed that more than one are rotating, half-rotation or while displaying that more than one are rotating The processing displayed that the player in game space does the throw of the throw-ed object by a certain actuation of a controller and the processing displayed that a throw-ed object flies in game space based on the above-mentioned throw energy, the direction of a throw, and an include angle are required. It is for bringing close the game held in game space by real game more visually.

[0005] However, if the direction to which a throw-ed object flies is decided with the posture of the player in game space, it will become difficult [a game player] for the player in game space to judge the timing which carries out the throw of the throw-ed object. It is in the condition of the player image by which it is indicated by sequential on a screen, and is because the player in game space has to judge the timing which carries out the throw of the throw-ed object.

[0006] Moreover, the prediction vector which is shown with the image of the arm of the player in game space in the case of the hammer throw differs from the prediction vector shown with the image of the hammer currently brandished by this player. For example, when the dynamic image of the player who does the throw of the throw-ed object is created using the technique of motion capture, inertia is expressed like a genuine article, even if visual. That is, the image of a hammer serves as this side from the image of a revolving player's arm in a hand of cut. Therefore, a game player grip-comes increasingly to be hard of the timing which makes the player in game space detaching a hammer.

[0007] Furthermore, a game which is made to carry out multiple-times rotation of the throw object, and carries out the throw of the throw-ed object like the hammer throw or the discus throw has a rotational frequency in use. Therefore, also in a game, when the rotational frequency whose rotational frequency of a throw object is in use is exceeded and the throw of the throw-ed object is not yet carried out, it is necessary to make it failure etc. It is also for securing game nature. However, counting looking at the condition that a throw object rotates what [after] rotation throw object I may rotate has a possibility of causing the fall of concentration to the game of a game player, as a result affecting the result of a game while a game player counts and causing a mistake.

[0008] This invention was made in consideration of such a point, and aims at guiding intelligibly the direction of a throw of a throw-ed object, and the remaining rotational frequencies to a game player.

[0009]

[Means for Solving the Problem] The image which one with main this invention shows the throw-ed object by which a throw is visually carried out to the image information which shows a throw object at least with the throw object concerned is displayed on the screen of a display means. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object visually, while being operated based on actuation of an actuation means Based on actuation of the above-mentioned actuation means, the throw of the above-mentioned throw-ed object is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means. It is the throw guide method of presentation in a throw game, and the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown is displayed.

[0010] Moreover, other one [main] of this inventions is the actuation or abbreviation rotation actuation on which actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object draws an arc in the above-mentioned invention.

[0011] Moreover, in the above-mentioned invention, a sequential indication of other one [main] of this inventions is given so that the above-mentioned throw guide image may draw

the orbit according to actuation of the above-mentioned throw object.
[0012] Moreover, in the above-mentioned invention, a sequential indication of other one [main] of this inventions is given so that the locus image in which the locus of the above-mentioned throw guide image is shown may draw the orbit according to actuation of the above-mentioned throw object.
[0013] Moreover, other one [main] of this inventions is generated in the above-mentioned invention based on the coordinate information on a throw guide image that the above-mentioned locus image was displayed as the coordinate information on one throw guide image before one rather than the throw guide image concerned.
[0014] Moreover, in the above-mentioned invention, as for other one [main] of this inventions, the false three-dimensional display of the above-mentioned throw guide image is carried out on the screen of the above-mentioned display means.
[0015] Moreover, other one [main] of this inventions is a configuration which has the visual direction directions function in which the configuration of the above-mentioned throw guide image is similar to an arrow-head configuration or this at least in the above-mentioned invention.
[0016] Moreover, in the above-mentioned invention, as for other one [main] of this inventions, adjustable [of the color of the above-mentioned throw guide image] is carried out at least according to the rotational frequency of the above-mentioned throw object.
[0017] Moreover, the color of the above-mentioned throw guide image changes from cold color to warm color according to being changed into the value from a value with the value of the rotational frequency of the above-mentioned throw object of the above-mentioned throw object small in the above-mentioned invention with other one [main / large] of this inventions.
[0018]
[Embodiment of the Invention] Below, sequential reference of drawing 1 - drawing 13 is carried out, and the gestalt of operation of this invention is explained at a detail.
[0019] Explanation of the gestalt of operation of this invention indicates the item explanation shown below at the head of each item, and explains it in the sequence shown below about each item.
[0020] A. Game structure of a system (drawing 1)
B. The function which CPU1 shown in drawing 1 has (drawing 2)
C. The display of an arrow head and a locus (they are drawing 4 C, D, G, and H to the drawing 3 list)
D. The example of image display (drawing 5 and drawing 6)
E. Control action by the main routine (drawing 7 - drawing 10)
F. Control action by the polygon image display routine S100 (drawing 11)
G. Control action by the throw guide display routine S200 (drawing 12 - drawing 14)
[0021] A. Game structure of a system (drawing 1)
[0022] Drawing 1 is the block diagram showing the example of the game structure of a system as a gestalt of 1 operation of this invention.
[0023] [Connection and configuration] The game system shown in this drawing 1 consists of a body of a game machine, and a record medium 30 with which the game data which become an image, voice, and a list from program data were recorded. The bus 2 by which the body of a game machine is set to CPU1 and this CPU1 from the address, data, and a control bus is connected. Into this bus 2 The graphics data generation processor 3, an interface circuitry 4, main memory 5, ROM6, the elongation circuit 7, a parallel port 8, a serial port 9, the drawing processing processor 10 and a buffer 11, the speech processing processor 13 And a buffer 14, a decoder 17 and a buffer 18, an interface circuitry 20, and memory 21 are connected, respectively. Furthermore, a television monitor 12 is connected to the drawing processing processor 10. A loudspeaker 16 is connected to the speech processing processor 13 through

an amplifying circuit 15, the record-medium driver 19 is connected to a decoder 17, a controller 22 is connected to an interface circuitry 20, and it is constituted.
[0024] Here, as for the above-mentioned game system, the gestalten differ according to an application. That is, when the above-mentioned game system is constituted as home use, a television monitor 12 and a loudspeaker 16 serve as a body of a game machine with another object. Moreover, when the above-mentioned game system is constituted as business use, all the components shown in drawing 1 are contained by one housing used as one apparatus. Moreover, when the above-mentioned game system is constituted as a nucleus, it sets a personal computer and a workstation. The above-mentioned television monitor 12 corresponds to the display for the above-mentioned computers. The above-mentioned drawing processing processor 10, the speech processing processor 13, and the elongation circuit 7 It corresponds to the hardware on the add-in board carried in some game program data currently recorded on the above-mentioned record medium 30, respectively, or the expansion slot of a computer. The above-mentioned interface circuitry 4, the above-mentioned parallel port 8, the above-mentioned serial port 9, and an interface circuitry 20 correspond to the hardware on the add-in board carried in the expansion slot of a computer. Moreover, the above-mentioned buffers 11, 14, and 18 correspond to each area of main memory 5 or the expanded memory which is not illustrated, respectively. With this gestalt, the above-mentioned game system explains taking the case of the case where it is constituted as home use.
[0025] Next, each component shown in drawing 1 is explained more to a detail. The graphics data generation processor 3 will play a role of a co-processor, if CPU1 says. That is, this graphics data generation processor 3 performs coordinate transformation and light source count, for example, the matrix of a fixed-point format and the operation of a vector, by parallel processing. The main processings of this graphics data generation processor 3 are coordinate transformation processing and light source computation. Coordinate transformation processing is processing which is due to movement magnitude data and rotation data in two-dimensional [of the image data supplied from CPU1], or the absolute-coordinate data of each top-most vertices within a three-dimension side, asks for the address on the display area of a processing-object image, and returns the address data concerned to CPU1 again. This coordinate transformation processing is explained in full detail behind.
[0026] Moreover, light source computation is processing which calculates the brightness of an image according to the vector data of a beam of light, the normal data showing the sense of the field of a polygon, and the data in which the color of a field is shown.
[0027] The above-mentioned interface circuitries 4 are objects for an interface, such as pointing devices, such as a peripheral device, for example, a mouse, and a trackball. The program data as an operating system of a game system are memorized by the above ROM 6. If it says with a personal computer, it is equivalent to BIOS (Basic Input Output System).
[0028] the intra based on MPEG (Moving Picture Engineering Group) or JPEG (Joint PictureEngineering Group) in the above-mentioned elongation circuit 7 -- elongation processing is performed to the compression image compressed by coding. elongation processing -- decoding (decoding of the data encoded by VLC:Variavle Length Code), reverse quantization processing, IDCT (Inverse Discrete Cosine Transform) processing, and intra -- it is restoration processing of an image etc.
[0029] The drawing processing processor 10 performs drawing processing to a buffer 11 based on the drawing instruction which CPU1 publishes. A buffer 11 consists of display area and non-display area. Display area is the expansion area of the data displayed on the screen of a television monitor 12. Non-display area is storage areas, such as texture data and color palette data. Here, texture data are two-dimensional image data. Color palette data are data for specifying colors, such as texture data. These data are divided and read from a record medium 30 to multiple times by CPU1 according to the advance situation of 1 time or a game,

and are memorized beforehand in the non-display area of a buffer 11.
[0030] There is a drawing instruction for drawing the drawing instruction for drawing a three-dimensional image as a drawing instruction using the drawing instruction for drawing Rhine, for example and a polygon and the usual two-dimensional image. Here, a polygon is a polygonal two-dimensional image and a triangle or a square is used in this gestalt.
[0031] The drawing instruction for drawing Rhine consists of data which mean drawing initiation and the ending address of Rhine, a color, and Rhine drawing. This Rhine drawing instruction is directly published to the drawing processing processor 10 by CPU1.
[0032] The drawing instruction for drawing a three-dimensional image using a polygon consists of the polygon top-most-vertices address data on the display area of a buffer 11, texture address data in which the storage location on the buffer 11 of the texture data stuck on a polygon is shown, and brightness data which show the brightness of a texture to the color palette address-data list which shows the storage location on the buffer 11 of the color palette data in which the color of texture data is shown. Polygon top-most-vertices address data are obtained among these data by calculating based on the data to which the graphics data generation processor 3 indicates a motion of a polygon to be polygon absolute-coordinate data from CPU1, and the data in which a motion of a view location is shown. Here, it explains that polygon top-most-vertices address data are called for how.
[0033] A motion of the body on the screen of a television monitor 12 is decided by the motion of a view location to a motion of objective the very thing and this body. For example, a motion of the body on the thing to which only a body moves and the view location is being fixed, then the screen of a television monitor 12 is a motion of the body itself. On the contrary, the motion on the thing from which there is no motion in a body and only the view location was moved, then the screen of a television monitor 12 is a motion of the view location itself. In addition, it will be easier to understand that a "view location" reads it as a "camera location." That is, a display which picturized the body is performed, moving a camera on the screen of a television monitor 12. Although the case where either a body or a view location moved was explained in order to simplify explanation, usually it is processed as both the body and the view location are moving, and the result is displayed.
[0034] Here, "a motion" of the above-mentioned body consists of a "rotation" and "movement magnitude." The rotation of the body to a view location is generated by the objective angle of rotation and the angle of rotation of a view location. Here, a rotation and an angle of rotation are expressed in procession of 3x3 by the processing for which 2x2 and three-dimension system of coordinates are used in the processing for which two-dimensional system of coordinates are used. Moreover, the movement magnitude of the body to a view location is generated by the objective location (coordinate value), the location (coordinate value) of a view location, and the angle of rotation of a view location. Here, an angle of rotation is expressed in procession of 3x3 like **** by the processing for which 2x2 and three-dimension system of coordinates are used in the processing for which two-dimensional system of coordinates are used. In addition, the objective angle of rotation based on actuation of a controller 22 and the angle of rotation of a view location are held as a table, respectively. Based on actuation of a controller 22, from the above-mentioned table, CPU1 reads the angle of rotation of a corresponding body or a view location, and it is used for it in order to calculate the rotation of a body [as opposed to a view location for the read angle of rotation], and movement magnitude.

[0035] The polygon top-most-vertices address data on display area are called for as follows so that the above explanation may show. That is, according to actuation of a controller 22, an objective angle of rotation and a location, and a list are asked for the angle of rotation of a view location, and a location by CPU1. Next, based on an objective angle of rotation and the angle of rotation of a view location, the rotation of the body to a view location is calculated by CPU1.

And based on an objective location, the location of a view location, and an angle of rotation, the movement magnitude of the body to a view location is calculated by CPU1. The rotation and movement magnitude data of these bodies are expressed with the matrix of 3x3 when three-dimension system of coordinates are used and processed, as already explained.

[0036] The rotation and movement magnitude data of the above-mentioned body are given to the graphics data generation processor 3 with the absolute-coordinate data of a polygon. The graphical data generation processor 3 changes the absolute-coordinate data of a polygon into polygon top-most-vertices address data based on the rotation and movement magnitude data of the above-mentioned body. The above is processing until polygon top-most-vertices address data are obtained.

[0037] The above-mentioned polygon top-most-vertices address data show the address on the display area of a buffer 11. The drawing processing processor 10 sets up the range of the triangle shown with 3 or four polygon top-most-vertices address data, or a square 1 on the display area of a buffer 11, and writes in the texture data corresponding to the range concerned. Generally this processing is called "A texture should stick" etc. On the screen of a television monitor 12, the body with which texture data were stuck on many polygons is displayed by this.

[0038] The drawing instruction for drawing the usual two-dimensional image consists of top-most-vertices address data, texture address data, and brightness data that show the brightness of a texture to a color palette address-data list. Top-most-vertices address data are coordinate data with which the graphics data generation processor 3 carries out coordinate transformation of the top-most-vertices coordinate data on the flat surface from CPU1, and obtains it based on the movement magnitude data from CPU1 among these data. Hereafter, it simplifies like "publishing a drawing instruction" and drawing processing is indicated.

[0039] The speech processing processor 13 makes a sound source the ADPCM data which memorized the ADPCM data read from the record medium 30 to the buffer 14, and were memorized by this buffer 14. And the speech processing processor 13 reads ADPCM data with a clock with a frequency of 44.1kHz. and -- and the speech processing processor 13 processes conversion of a pitch, addition of a noise, setup of an envelope, setup of level, addition of RIBABU, etc. to the ADPCM data read from the buffer 14. When the voice data read from a record medium 30 is PCM data, this PCM data is changed into ADPCM data by CPU1. Moreover, processing by the program data to PCM data is directly performed on main memory 5. By being processed on main memory 5, further, the voice data encoded by the data of an ADPCM format is outputted from a loudspeaker 16 as voice, after various processings which the speech processing processor 13 was supplied and were mentioned above are performed.

[0040] The record-medium drivers 19 are a hard disk drive, an optical disk drive, a flexible disk drive, a silicon disk drive, a cassette medium readout machine, etc. Record media 30 are a hard disk, an optical disk, a flexible disk, semiconductor memory, etc. From a record medium 30, the record-medium driver 19 reads an image, voice, and game program data, and supplies the read data to a decoder 17. A decoder 17 supplies the data which performed error correction processing by ECC (Error Correction Code), and performed error correction processing to main memory 5 or the speech processing processor 13 to the playback data from the record-medium driver 19.

[0041] Memory 21 consists of memory of a holder and a card mold. The memory of a card mold is for holding the various parameters of a game like holding the condition for example, at the termination time. A controller 22 consists of cross-joint key [which consists of the left key L, the right key R, an upper key U, and a bottom key D], left carbon button 22L and right carbon button 22R, start button 22a, select button 22b, 1st carbon button 22c, 2nd carbon button 22d, 3rd carbon button 22e, and 4th carbon button 22f. A cross-joint key gives the

command with which a game player shows four directions to CPU1. Start button 22a is for a game player to direct initiation of the game program data loaded from a record medium 30 to CPU1. Select button 22b is for a game player to direct the various selections about the game program data loaded to main memory 5 from a record medium 30 to CPU1. In addition, left key 22L, right key 22R, and the 1st - carbon button [4th /c / 22 /,d / 22 /,e / 22 /, and 22f] function change with game program data loaded from a record medium 30.

[0042] [Actuation] An electric power switch (not shown) is turned ON and a power source is supplied to a game system. If the record-medium driver 19 is loaded with the record medium 30 at this time, CPU1 directs read-out of the game data from a record medium 30 to the record-medium driver 19 based on the operating system memorized by ROM6. Thereby, the record-medium driver 19 reads an image, voice, and game program data from a record medium 30. The image, voice, and game program data which were read are supplied to a decoder 17, and error correction processing is performed here. The image data to which error correction processing was performed in the decoder 17 is supplied to the elongation circuit 7 through a bus 2, after elongation processing mentioned above is performed here, is supplied to the drawing processing processor 10, and is written in the non-display area of a buffer 11 by this drawing processing processor 10.

[0043] The voice data with which error correction processing was performed in the decoder 17 is supplied to main memory 5 or the speech processing processor 13, and is written in main memory 5 or a buffer 14. Moreover, the game program data with which error correction processing was performed in the decoder 17 are supplied to main memory 5, and are written in this main memory 5. Henceforth, CPU1 advances a game based on the contents which a game player directs through a controller 22 in the game program data memorized by main memory 5 and a list. That is, CPU1 performs control of an image processing, control of speech processing, and control of internal processing suitably based on the contents of directions directed from a game player through a controller 22. Control of an image processing is supply to the graphics data generation processor 3 of the rotation and movement magnitude data which were mentioned above, or absolute-coordinate data, issue of the drawing instruction containing address data and brightness data on the display area of the buffer 11 for which the graphics data generation processor 3 asked, etc. Control of speech processing is assignment of issue of the voice output command to the speech processing processor 13, level, RIBABU, etc. Control of internal processing is an operation according to actuation of a controller 22 etc.

[0044] B. The function which CPU1 shown in drawing 1 has (drawing 2)

[0045] Drawing 2 is the explanatory view showing the function which CPU1 shown in drawing 1 has. CPU1 is read from the record medium 30 shown in drawing 1 , and has the function shown in drawing 2 by reading the program data memorized by main memory 5. The function of CPU 1 shown in this drawing 2 consists of information setting means 1e, 1f of decision means, 1g of drawing instruction issue means, 1h of variable-setting means, number acquisition means of coma 1i, migration and rotation acquisition means 1j, polygon information-management means 1k, and 1 m of guide information-management means as a result of button-grabbing detection means 1a, view location data setting means 1b, display-rectangle information extract means 1c, and 1 d of operation means. These means serve as a subject of control who explains in Item E - Item G, respectively.

[0046] C. The display of an arrow head and a locus (they are drawing 4 C, D, G, and H to the drawing 3 list)

[0047] Drawing 3 A for example, the explanatory view for explaining the method of a display of the throw guide which consists of the arrow head and locus on the screen which consists of 480 pixels long and 640 pixels wide and drawing 3 B The explanatory view showing the absolute coordinate of the image of an arrow head and drawing 3 C are the explanatory views for explaining notionally the image deformation processing when setting up an arrow head on a

three-dimension coordinate based on an absolute coordinate, and the movement magnitude and rotation data of the arrow head shown in drawing 3 B. Moreover, the explanatory view in which drawing 4 C shows an example of an arrow-head absolute coordinate and drawing 4 D The arrow-head absolute-coordinate data shown in drawing 4 C are based on movement magnitude and rotation data. Explanatory view and drawing 4 G which shows an example of the arrow-head address after conversion changed by the graphics data generation processor 3 shown in drawing 1 The explanatory view showing an example of the arrow-head posterior part address after conversion which is equivalent to the tail of an arrow head among the arrow-head addresses after conversion shown in drawing 4 D, and drawing 4 H are the explanatory views showing an example of the address for locus generation obtained from the arrow-head posterior part address after conversion shown in drawing 4 G.

[0048] Let it be the purpose to guide the remaining rotational frequencies of a throw object to the timing to which the throw object in game space used as the direction of a throw which a game player means carries out the throw of the throw-ed object, and a list intelligibly for a game player in this gestalt. This purpose is attained by displaying the locus Lo of the arrow head Na which shows the direction where a throw object carries out the throw of the throw-ed object, and this arrow head Na on the screen of a television monitor 12, as shown in drawing 4 A. And the direction which an arrow head shows is in agreement in the direction of a throw of the throw-ed object which the posture of throw objects, such as a player displayed with the arrow head, shows. That is, the arrow head which shows the direction of a throw according to the posture of the throw object displayed every moment on the screen is displayed every moment with a throw object.

[0049] Moreover, an arrow head Na is displayed in order toward the direction shown by Orbit OR top and the arrow head. At this time, two address data a1 and a2 of the part of the tail of the arrow head Na by which it is indicated by current, and two address data b2 and b1 of the part of the tail of the arrow head Nb displayed before one are used, and one image is generated and displayed. This image is the locus image Lo used as a locus of an arrow head Na. As shown in this drawing 3 A, the locus image Lo is constituted by two triangles formed by tying the address data a1 of the part of the tail of an arrow head Na, and the address data b2 of the part of the tail of an arrow head Nb, and tying the address data a2 of the part of the tail of an arrow head Na, and the address data b1 of the part of the tail of an arrow head Nb.

[0050] And the rotational frequency of a throw object is further expressed by the color of an arrow head Na. For example, the color of the arrow head Na when there are few values which show a rotational frequency is set as purple, the color of an arrow head Na is set as warm color one by one as the value which shows a rotational frequency becomes large, and the color of the arrow head Na when the value which shows a rotational frequency is the largest is set as red. As for an arrow head, in the case of the shot put, a throw game is displayed in red from the beginning. It is because rotational frequencies in use are 0.5 rotations in a shot put game. Moreover, as for an arrow head, in the case of the discus throw, a throw game is displayed [yellow and 2 rotation eye] in red by 1 rotation eye, respectively. It is because rotational frequencies in use are two rotations in a discus throw game. moreover, a throw game -- the case of the hammer throw -- an arrow head -- in 1 rotation eye, it is displayed [blue and 3 rotation eye / green and 4 rotation eye / yellow and 5 rotation eye] in red by purple and 2 rotation eye, respectively. Although rotational frequencies in use are four rotations in a hammer throw game, it is because it is considering as five rotations with this gestalt. In addition, on processing, based on the division value (below decimal point cut-off) acquired by 1d of operation means doing the division of the data Ah by 360 whenever [throw azimuth], the color of an arrow head is called for, when 1g of drawing instruction issue means refers to a table. A table consists of data in which the color palette address data or the color corresponding to many division values and the division value of these large number is shown,

respectively.
[0051] Moreover, in this gestalt, when the maximum above-mentioned rotational frequency is exceeded, it considers as failure. This mistake is made in the foul in a real game.
[0052] The above throw guide displays are expressed as real time according to the posture of a throw object. Therefore, a game player can know the direction of a throw and the remaining rotational frequency when a throw object carries out the throw of the throw-ed object at present. Therefore, by the above-mentioned throw guide display, a game player can recognize the direction of a throw and the timing to mean of the throw-ed object which he means, can operate a controller 22 based on this recognition, and can carry out the throw of the throw-ed object to a throw object. That is, the throw game excellent in the user interface is realizable. Hereafter, the processing for carrying out the above-mentioned guide display to a detail is explained more.
[0053] An arrow head is a configuration as shown for example, in drawing 3 A. And the absolute coordinate of this arrow head is the coordinate data of the angle of the quadrilateral surrounding the arrow head on the three-dimension flat surface containing the arrow head at the time of setting a core to "0", as shown in drawing 3 A. Moreover, this arrow-head absolute-coordinate data is data loaded to main memory 5 from the record medium 30. In the example of the arrow-head absolute-coordinate data shown in this drawing 3 B and drawing 4 B (x1, y1, z1) -- x2, and (y2, z2) -8, (0, 10), and (x3, y3, z3) -- (8, 0, -10), and (x4, y4, z4) -- (- it is 8, 0, and -10). [(8, 0 10), and]
[0054] Drawing 3 C shows notionally transform processing by the graphics data generation processor 3. If the arrow-head absolute-coordinate data shown in drawing 3 R>3B, the movement magnitude mentioned above, and rotation data are given from CPU1, the graphics data generation processor 3 will set up an arrow head on a three-dimension coordinate, as shown in drawing 3 C. The location and angle of rotation of a camera shall be fixed to the location of explanation a throw object and a throw-ed object always appear for convenience.
[0055] The address data of the top-most vertices of the quadrilateral surrounding the arrow head set up on the three-dimension coordinate are changed into the address data on the display area of a buffer 11. The example of this arrow-head address after conversion is shown in drawing 4 D. The arrow-head address after conversion is the address data on the display area of buffer memory 11, as shown in drawing 4 D. for example, the addresses a1, a2, a3, and a4 on the display area of the buffer 11 of the angle of the quadrilateral surrounding the arrow head Na at present shown in drawing 3 A -- respectively -- (560, 320), and (544, 324) -- it is [and / (560 394) / (594 368)]. moreover, the addresses b1, b2, b3, and b4 on the display area of the buffer 11 of the angle of the quadrilateral which surrounds the arrow head Nb in front of one from this time -- respectively -- (424, 342), and (408, 346) -- it is [and / (424 416) / (458 390)] (refer to drawing 3 A and drawing 4 D).
[0056] Next, the locus image Lo is explained. It explains as that whose arrow head which the arrow head Na was displayed at present, and the arrow head Nb was displayed before predetermined unit time amount, and was displayed in the past is only an arrow head Nb.
[0057] As shown in drawing 3 A, when the arrow head Na is displayed, the arrow-head address data after conversion of the arrow head Na currently displayed at present as shown in drawing 4 D, and the address data after conversion of the arrow head Nb currently displayed from this time at the time before one are memorized by main memory 5. On the other hand, as shown in drawing 4 G, only the address which is equivalent to the tail of an arrow head among the arrow-head addresses after conversion shown in drawing 4 D is memorized by main memory 5 as after [conversion] arrow-head posterior part address data. As it turns out that drawing 3 A is referred to, the addresses for a tail of the arrow head Nb displayed the arrow head Na currently displayed at present and one frame ago are a1, a2, and b1 and b2, respectively. And these values are (560, 320), and (544 324) (424 342) (408, 346) as shown in

the lower part of this drawing 3 A. These values are memorized as the after [conversion] arrow-head posterior part address in other area of main memory 5 as shown in drawing 4 R>4G.

[0058] And as address data for locus generation, after the arrow-head posterior part address data after both conversion of arrow heads Na and Nb are memorized by main memory 5, they are supplied to the drawing processing processor 10, as shown in drawing 4 H. The drawing processing processor 10 will write in the texture data of an arrow head as locus image data on the display area of a buffer 11 based on these address data, if the drawing instruction containing the above-mentioned address data for locus generation is received. Thereby, the locus image Lo is displayed on the screen so that a slash may be attached and shown in drawing 3 A. This locus image Lo appears visual just like the locus of an arrow head Na. And the orbit OR of an arrow head is visually expressed by giving a sequential indication of an arrow head and the locus.

[0059] Since the number of the arrow heads displayed on this time and the past is two and the locus image Lo displayed also serves as a large number in this example when this number is a large number although the number of the locus images Lo displayed is one, the die length of the locus image Lo displayed on the arrow head currently displayed on the screen by following becomes long.

[0060] If the orbit OR of an arrow head is expressed by the display of the locus image Lo, since the location where an arrow head is displayed beforehand understands a game player, only in the part, its response to the direction of a throw shown by the arrow head will improve. That is, the direction where an arrow head is displayed on the orbit OR currently expressed can predict the display position of an arrow head rather than an arrow head is displayed on the place which does not have anything one by one, and an actuation reaction when the arrow head which should carry out a throw to a throw object by this is displayed can be raised.

[0061] Next, the color of an arrow head Na is explained. As already explained, with this gestalt, the color of an arrow head Na is changed according to rotation of a throw object. This was already explained. Usually, cold color is used in order to express a thing small like the lowness of temperature. Moreover, warm color is used in order to express the large thing like the height of temperature. In this gestalt, the color of an arrow head Na is changed into warm color from cold color one by one according to a rotational frequency increasing. Therefore, a game player can recognize the remaining rotational frequencies. In other words, a game player can recognize the timing which carries out the throw of the throw-ed object to the throw object in game space. Moreover, red is usually used as a color expressing "risk", a "limitation", etc. Therefore, if the color of an arrow head is brought close to red as a rotational frequency rises, a game player can recognize certainly that the remaining rotational frequencies decrease.

[0062] D. The example of image display (drawing 5 and drawing 6)

[0063] Drawing 5 A - drawing 5 D is the explanatory view showing the example of a screen display in a hammer throw game. The player Ma in game space extracts four examples of a screen display of Uchi of many a series of frames after starting the actuation for throwing Hammer Ba until it finishes throwing Hammer Ba, and shows serially this drawing 5 A - drawing 5 D. Serial sequence is drawing 5 A, drawing 5 B, drawing 5 C, and drawing 5 D. The example of a screen display at the time of drawing 5 D throwing Hammer Ba for the example of a screen display which is performing rotation actuation just before drawing 5 C throws Hammer Ba for the example of a screen display at the time of starting rotation actuation for drawing 5 B throwing Hammer Ba for the example of a screen display at the time of starting actuation for drawing 5 A throwing Hammer Ba is shown. In addition, a sign gives a sign only to drawing 5 B for convenience. Moreover, each example of a screen display is chosen from a full screen when a multiple-times game is performed.

[0064] One screen consists of a window W for displaying the image for a display as a result of

a player's besides a background image Ma, the image Ba of a hammer, the image Na of an arrow head, and the locus image Lo, and a guide image Gi for an include-angle display as shown in drawing 5 B. Moreover, Lines LL and LR are displayed into a background image. The alphabetic character (this example 1 P:1 person) which shows the number of game players is displayed on the upper part of the left end part of Window W, the bar graph which shows throw energy is displayed on that lower part, and the alphabetic character (this example PLAYER1) which shows the how many persons' game player it is is further displayed on that lower part. Moreover, the alphabetic character of "1ST", "2ND", and "3RD" is displayed in the center of Window W, respectively. These are "the 1st time", "the 2nd time", and "the 3rd semantics", respectively. And the alphabetic character which shows flight distance or foul is displayed on the right-hand of the alphabetic character of the above "1ST", "2ND", and "3RD." It is shown by drawing 5 A that the flight distance in the 1st throw is [the flight distance in "59.33M" (59m 33cm is meant) and the 2nd throw] "68.39M" (68m 39cm is meant). In addition, "x" shown in the right-hand side of the alphabetic character which shows the flight distance of "1ST", and "O" shown in the right-hand side of the alphabetic character which shows the flight distance of "2ND" to a list show that the flight distance in "2ND" is an effective value rather than the flight distance in "1ST." Moreover, in drawing 5 B, it is shown by the display of "FOUL x" that the 1st throw is failure, i.e., foul.

[0065] Moreover, it is displayed that it becomes small, so that the locus image Lo is so large that it is close to an arrow head and it is far as shown in drawing 5 B.

[0066] E. Control action by the main routine (drawing 7 - drawing 10)

[0067] Drawing 6 - drawing 9 are the flow charts for explaining the control action by the main routine of a throw game. A throw game says that a throw-ed object carries out a throw into game space with the throw object in game space according to actuation of the controller 22 by the game player. As a throwing event, game, for example, the shot put, that rotation stripes carry out [a throw object] the throw of the **** throw object, the discus throw, and the hammer throw are mainly known. Therefore, in the case of a shot put game, the throw game shown in drawing 6 - drawing 9 sets. Read a "throw object" with a "player", read a "throw-ed object" with a "cannonball", change, and, in the case of a discus throw game, it sets. A "throw object" is read with a "player", a "throw-ed object" is read with a "disk", and it changes, and in the case of a hammer throw game, a "throw object" shall be read with a "player", a "throw-ed object" shall be read with a "hammer", and it shall change to it.

[0068] In addition, it is the control action by the operating system memorized by ROM6 which showed only step S1 to drawing 1 . Other steps are the control action by the game program data read from the record medium 30. Moreover, the subject of control by game program data is each means as a function of CPU1 shown in drawing 2 , as already explained.

[0069] At step S1, the record-medium driver 19 reads an image, voice, and game program data from a record medium 30 with the instruction of an operating system. Program data are memorized by main memory 5 among the read data. Thereby, CPU1 has the function shown in drawing 2 . In addition, at this time, an image, i.e., texture data, shall be memorized in the non-display area of the buffer 11 of the drawing processing processor 10, and a texture number shall be assigned, respectively. Moreover, voice data shall be memorized by the buffer 14 of the speech processing processor 13, and voice number data shall be assigned, respectively. Usually, although all images and voice data are not held in step S1 at buffers 11 and 14, all images and voice data consider as the thing of explanation loaded in step S1 for convenience. At step S2, it judges whether it is the no on which start button 22a of a controller 22 was pushed, and if button-grabbing detection means 1a is "YES", it will shift to step S3.

[0070] At step S3, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of a selection image is shown to the drawing processing processor 10 shown in drawing 1 . The drawing processing processor 10 develops the image data of a selection

image on the screen of a buffer 11 based on the above-mentioned drawing instruction. Thereby, a selection image is displayed on the screen of a television monitor 12. In step S4, it judges whether start button 22a of a controller 22 was pushed, and if button-grabbing means 1a is "YES", it will shift to step S5.

[0071] CPU1 sets to the selected game at step S5. Here, with reference to the selection image with which the game player was displayed ["it is selected" and] at step S3, a game is chosen using a cross-joint key and it means pushing start button 22a next. Moreover, the character in other, for example, waging war, mold fighting game [game / itself] etc. is included with a "game" here. In short, it is the selection matter before a game is actually started. In this step S5, it considers as the thing of explanation as which the throw game was chosen for convenience. At step S6, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of the initial image of the selected game is shown to the drawing processing processor 10. Thereby, the drawing processing processor 10 writes in the image data of an initial image on the display area of a buffer 11. Thereby, an initial image is displayed on the screen of a television monitor 12.

[0072] At step S7, 1h of variable setting means resets the flag and variable which are held to main memory 5, respectively. At step S8, it judges whether 1st carbon button 22c was pushed, if it is "YES", it will shift to step S9, and if button-grabbing detection means 1a is "NO", it will shift to step S11. Actuation of 1st carbon button 22c is for controlling the rotational speed of the throw object in game space.

[0073] In step S9, 1d of operation means adds the reference speed data s to the speed data Sd at the time of a rise.

[0074] At step S10, number acquisition means of coma 1i asks for the number data fd of coma according to the value of the speed data Sd. This number data fd of coma is called the number data of conversion coma. The number data fd of conversion coma are called for by referring to a table. A table consists of much speed data Sd and number data of coma of a large number registered about the speed data Sd of these large number, respectively, and is loaded to main memory 5 from a record medium 30. In addition, you may make it ask for the number of coma by performing a predetermined operation using the speed data Sd.

[0075] At step S11, 1d of operation means subtracts the reference speed data m from the speed data Sd at the time of descent. At step S12, if the value of the speed data Sd judges whether it is minus and is "YES", it will shift to step S13, and if 1f of decision means is "NO", they will shift to step S10.

[0076] At step S13, 1h of variable setting means substitutes "0" for the speed data Sd. When button-grabbing detection means 1a judges whether 1st carbon button 22c was pushed at the above-mentioned step S8 and it judges that 1st carbon button 22c is pushed When it is judged as that on which shift to step S9, the speed data Sd are made to increase, and 1st carbon button 22c is not pushed It is for setting up the value of the speed data Sd according to the count of press of 1st carbon button 22c per unit time amount by the game player to shift to step S11 and to decrease the speed data Sd. That is, if there are many counts by which a game player pushes 1st carbon button 22c into unit time amount, it will go up, the rotational speed, i.e., the throw energy, of a throw object in game space. On the other hand, if the count by which a game player pushes 1st carbon button 22c into unit time amount is low, it will descend, the rotational speed, i.e., the throw energy, of a throw object in game space. Although not shown as a step in the flow chart, the display of the above-mentioned throw energy is expressed by change of a bar graph shown in drawing 5 A - drawing 5 D.

[0077] At step S14, 1d of operation means adds the number data fd of conversion coma to the number data FD of coma. At step S15, number acquisition means of coma 1i reads Data Ah from the table TBL currently held at main memory 5 whenever [corresponding to the number data FD of coma called for in step S14 / throw azimuth]. Here, Table TBL consists of data Ah

whenever [throw azimuth / of a large number registered, respectively about much number data FD of coma, and the number data FD of coma of these large number] as shown in drawing 4 A.

[0078] At step S16, 1h of variable setting means reads the movement magnitude data according to a value and rotation data of the address data on the main memory 5 of the absolute-coordinate data of the polygon of a throw object, and the number data FD of coma from a table, the above-mentioned movement magnitude data are substituted for the movement magnitude variable MO, and the above-mentioned rotation data are substituted for the rotation variable RO for the above-mentioned address data at the address variable ADD, respectively. Here, rotation data are obtained by the above-mentioned address data and the movement magnitude data list by referring to a table. This table becomes the address data of the number of coma from the minimum value to maximum, and a large number registered according to the value of these numbers of coma, respectively, and a movement magnitude data list from rotation data.

[0079] Polygon image display processing is performed at step S100. This polygon image display routine S100 is explained in full detail behind. Throw guide display processing is performed at step S200. This throw guide display routine S200 is explained in full detail behind.

[0080] At step S17, it judges whether it is below the value of the minimum value data Ahmin of whenever [throw azimuth], whenever [throw azimuth], if the value of Data Ah is "YES", it will shift to step S18, and if 1f of decision means is "NO", they will shift to step S22. Here, the minimum value data Ahmin of whenever [throw vectorial angle] are the smallest include angle and an include angle to which a throw-ed object falls in un-effective area with allowances, as shown in drawing 4 A. Here, un-effective area is area out of range shown among Lines LL and LR as shown in drawing 4 B, drawing 5 A - drawing 5 D, respectively.

[0081] By the shot put, the maximum engine speed at the time of the throw for which it opts in this gestalt is become to "2" by "0.5" and the discus throw, and has become "5" in the hammer throw. Then, in this gestalt, the maximum of the number of coma is decided according to the above-mentioned maximum engine speed. For example, if the value of the number of the maximum coma shall be set as "240" for the rotational frequency set up by ****** by 4 times, the value of the number of the maximum coma in the game whose rotational frequency is 2 times will be set to "120." Moreover, in the above-mentioned track and field, "failure" means "foul."

[0082] In step S17, when 1f of decision means judges it as "NO", it shifts to step S18 and result information setting means 1e supplies the alphabetic data in which "FOUL x" is shown to the drawing processing processor 10 at this step S18. Thereby, as shown in drawing 5 B, the alphabetic character which shows "FOUL x" is displayed on the central part of Window W as an image. At step S300, throw object image display processing for failure is performed. This throw object image display routine S300 for failure consists of processing performed at setting processing and step S100 which are performed at step S32 mentioned later. On the screen, a motion of the throw object at the time of failure is expressed, for example like a throw object collapsing.

[0083] At step S19, it judges whether which carbon button was pushed by the existence of the data which 1f of decision means shows the carbon button pushed from button-grabbing detection means 1a, if it is "YES", it will shift to step S20, and if it is "NO", it will shift to step S300 again. At step S20, 1d of operation means adds "1" to the count data Th of a throw.

[0084] At step S21, it judges whether the count data Th of a throw are larger than the maximum Thmax of the count data of a throw, if it is "YES", it will shift to step S3 again, and if 1f of decision means is "NO", they will shift to step S8 again. At step S22, it judges whether 2nd carbon button 22d was pushed, if it is "YES", it will shift to step S23, and if button-grabbing

detection means 1a is "NO", it will shift to step S8 again. Here, throw timing is determined as the throw vertical include-angle data Av of a throw-ed object 2nd carbon button 22d. With this gestalt, while 2nd carbon button 22d is pushed, the sequential increment of the value of the throw vertical include-angle data Av is carried out. In addition, the value of the above-mentioned throw vertical include-angle data Av is displayed on real time with Guide Gi, as shown in drawing 5 A - drawing 5 D.

[0085] At step S23, 1d of operation means adds the criteria include-angle data z to the throw vertical include-angle data Av. At step S24, it judges whether it is larger than the value of the maximum data Avmax of the throw vertical include-angle data Av, if the value of the throw vertical include-angle data Av is "YES", it will shift to step S25, and if 1f of decision means is "NO", they will shift to step S26.

[0086] At step S25, 1h of variable setting means substitutes the maximum data Avmax of the throw vertical include-angle data Av for the throw vertical include-angle data Av. At step S26, it judges whether 2nd carbon button 22d was detached, if it is "YES", it will shift to step S27, and if button-grabbing detection means 1a is "NO", it will shift to step S23 again. When 2nd carbon button 22d is detached, the throw of the throw-ed object is carried out with the throw object in game space with the value of the throw vertical include-angle data Av at the time.

[0087] At step S27, 1h of variable setting means initializes the rate vector data of a throw-ed object based on Data Ah and the value of the throw vertical include-angle data Av the value of the speed data Sd, and whenever [throw azimuth]. Here, rate vector data shows the location on a three-dimension coordinate, and consists of x, and (y, z). Here, "initialization" means setting up the rate vector data decided by the three above-mentioned values. At step S28, 1d of operation means adds gravitational acceleration vector data to the rate vector data initialized at step S27. Here, gravitational acceleration vector data is a constant for changing the location on the three-dimension coordinate which the above-mentioned rate vector data shows (x y, z).

[0088] At step S29, 1d of operation means adds rate vector data (x y, z) to the location data (x y, z) of a non-throw object. Here, the location data of a non-throw object show the location of the throw-ed object on a three-dimension coordinate, and consist of x, and (y, z). At step S30, if the height of a throw-ed object judges whether it is "0" and is "YES", it will shift to step S31, and if 1f of decision means is "NO", they will shift to step S32. Here, height means the height on the screen of the throw-ed object for every frame.

[0089] At step S31, 1h of variable setting means sets the height of a throw-ed object to "0." step S32 -- 1h of variable setting means -- the above-mentioned movement magnitude data are substituted for the movement magnitude variable MO, and the above-mentioned rotation data are substituted for the address variable ADD for the address data on the main memory 5 of the absolute-coordinate data of the polygon of a throw-ed object at the rotation variable RO, respectively. Here, rotation data are obtained by the above-mentioned movement magnitude data list by referring to a table. This table becomes the movement magnitude data list of a large number registered according to the value of the location data of the throw-ed object from the minimum value to maximum, and these location data, respectively from rotation data. Of course, based on the above-mentioned location data, you may ask for the above-mentioned movement magnitude data and rotation data by the operation at any time.

[0090] Polygon image display processing is performed at step S100. At step S400, after [a throw] throw object image display processing is performed. This throw object image display routine S400 is a routine which consists of processing performed at the processing performed at step S32, and step S100. The motion after a throw object carries out the throw of the throw-ed object on the screen is expressed.

[0091] At step S33, 1d of operation means finds flight distance. And result information setting means 1e supplies the alphabetic data in which flight distance is shown to the drawing processing processor 10. At step S34, 1g of drawing instruction issue means supplies the

drawing instruction in which the display of the image for a result display is shown to the drawing processing processor 10. On the screen of a television monitor 12, the alphabetic character which shows the flight distance of a throw-ed object is displayed on the central part of the window W shown in drawing 5 A - drawing 5 D as an image by this.
[0092] At step S35, based on the existence of the data which 1f of decision means shows the manual operation button from button-grabbing detection means 1a, it judges whether which carbon button was pushed, and if it is "YES", it will shift to step S36. At step S36, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of the image at the time of a success is shown to the drawing processing processor 10. Thereby, the drawing instruction processor 10 writes in the image data at the time of a success on the display area of a buffer 11. Therefore, the image at the time of a success is displayed on the screen of a television monitor 12. Here, the image at the time of a success is an image of alphabetic characters, such as "GOOD", etc., for example, it is displayed on the central part of the window W shown in drawing 5 A - drawing 5 D.
[0093] At step S500, throw object image display processing is performed at the time of a success. The throw object image display routine S500 is a routine which consists of processing performed at the processing performed at step S32, and step S100 at the time of this success. On the screen, a motion of the throw object at the time of a success is expressed like a throw object flying and being over, for example. Replay image display processing is performed at step S600. The actuation information on the controller 22 according [this replay image display processing] to a game player is used. That is, the actuation information on the controller 22 by the game player is memorized in detail by main memory 5. And in this replay image display processing, sequential processing is performed based on the above-mentioned actuation information. Thereby, the image display condition based on the actuation which the game player performed is reproducible.
[0094] At step S37, it judges whether which carbon button was pushed by the existence of the data which 1f of decision means shows the contents of actuation from button-grabbing detection means 1a, and if it is "YES", it will shift to step S20 again.
[0095] F. Control action by the polygon image display routine S100 (drawing 11)
[0096] Drawing 10 is a flow chart for explaining the control action by the polygon image display routine S100. By this polygon image display routine S100, polygon display processing of a throw object or a throw-ed object is performed. One throw object and throw-ed object consist of many polygons. As shown in drawing 4 R>4E, the polygon is memorized by main memory 5 as absolute-coordinate data (x y, z) of much polygon top-most vertices. And these absolute-coordinate data (x y, z) are changed into the polygon address data after conversion on the two-dimensional flat surface shown in drawing 4 F (x y) by the graphics data generation processor 3 based on the movement magnitude data and rotation data which were mentioned above. The polygon address data after conversion (x y) shown in this drawing 4 F are supplied to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. The drawing processing processor 10 writes in texture data on the display area of a buffer 11 based on the above-mentioned polygon address data after conversion based on a drawing instruction. Thereby, on the screen of a television monitor 12, the throw object and throw-ed object which consist of many polygons are displayed.
[0097] At step S101, polygon information management means 1k reads the absolute-coordinate data (x y, z) of the top-most vertices of the polygon on the main memory 5 which the value of the address variable ADD shows from main memory 5. At step S102, polygon information management means 1k supplies the data of the absolute coordinate of the top-most vertices of a polygon, the movement magnitude substituted for the movement magnitude variable MO, the rotation substituted for the rotation variable RO, the vector of a beam of light, and the normal of a polygon to the graphics data generation processor 3, respectively. the

graphics data generation processor 3 -- each above-mentioned data -- being based -- the polygon address data after conversion (x y) -- and it asks for brightness data, respectively and these data are supplied to polygon information management means 1k, respectively. At step S103, polygon information management means 1k writes the after [conversion] polygon address data (x y) and brightness data from the graphics data generation processor 3 in main memory 5, respectively.

[0098] At step S104, it judges whether all the absolute-coordinate data of the top-most vertices of a polygon were changed into the polygon address data after conversion, if it is "YES", it will shift to step S105, and if 1f of decision means is "NO", they will shift to step S102 again. At step S105, 1g of drawing instruction issue means reads after [conversion] address data (x y), and brightness data from main memory 5, and they supply after [the conversion concerned] address data (x y), and brightness data to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. Thereby, the drawing processing processor 10 writes in the texture data of a throw object on the display area of a buffer 11 based on the above-mentioned address data after conversion (x y). Therefore, on the screen of a television monitor 12, image data, such as a throw object which consists of many polygons, and a throw-ed object, is displayed.

[0099] At step S106, it judges whether all data were transmitted, if it is "YES", it will escape from this polygon image display routine S100, and if 1f of decision means is "NO", they will shift to step S105 again.

[0100] G. Control action by the throw guide display routine S200 (drawing 12 - drawing 14)

[0101] Drawing 11 - drawing 13 are the flow charts for explaining the control action by the throw guide display routine. A throw guide consists of an image of an arrow head, and a locus image in which the locus of this arrow head is shown.

[0102] At step S201, 1m of guide information management means reads the absolute-coordinate data of the top-most vertices of the quadrilateral surrounding an arrow head from main memory 5. At step S202, migration and rotation acquisition means 1j obtain the movement magnitude data and rotation data on the three-dimension coordinate of the quadrilateral surrounding an arrow head according to the distance data from the center of rotation O indicated to be the value of Data Ah to drawing 3 A whenever [throw azimuth]. Here, the above-mentioned distance data are a fixed value.

[0103] At step S203, 1m of guide information management means supplies the absolute-coordinate data, movement magnitude data, and rotation data of an arrow head to the graphics data generation processor 3. Thereby, the graphics data generation processor 3 changes the absolute-coordinate data of an arrow head on a three-dimension coordinate based on movement magnitude data and rotation data, from the coordinate data obtained by this conversion, obtains the address data after conversion on two-dimensional (x y), and supplies the address data after this conversion (x y) to 1m of guide information management means. At step S204, 1m of guide information management means writes the address data after conversion from the graphics data generation processor 3 (x y) in main memory 5.

[0104] At step S205, 1g of drawing instruction issue means reads the address data after conversion (x y) from main memory 5, and they supply the address data after the conversion concerned (x y) to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. In addition, 1g of drawing instruction issue means is acquired based on the operation value to which 1d of operation means carries out the color palette address data for displaying an arrow head by the color according to an engine speed. An operation value is calculated here by doing the division of the data Ah by 360 (degree) whenever [throw azimuth]. Thereby, the drawing processing processor 10 writes in the texture data of an arrow head on the display area of a buffer 11 by the color specified by the color palette based on the address data after conversion (x y). In addition, the decision of a

color is made by reference of a table. A table consists of the data or color palette address data for specifying the color registered, respectively about the division values and these multiplication values from the minimum value calculated by 1d of operation means to maximum.

[0105] At step S206, if the value of an address pointer P judges whether it is start-address Pstart and is "YES", it will shift to step S207, and if 1f of decision means is "NO", they will shift to step S209. Here, Pstart is a start address in the area where the address data after conversion are memorized. At step S207, 1d of operation means adds the number k of base addresses to an address pointer P. Here, the value of the number k of base addresses is capacity value required to memorize two address data.

[0106] At step S208, 1m of guide information management means memorizes the address data after conversion of the flat surface corresponding to the tail of an arrow head (x y) in the area which the value of the address pointer P on main memory 5 shows. And it escapes from this throw guide display routine S200. At step S209, it judges whether the value which an address pointer P shows is smaller than the value which the subtraction result of having reduced the number k of base addresses indicates to be start-address Pstart from an addition result with the number nmax of the maximum storage of the address data after conversion, if 1d of operation means is "YES", they will shift to step S210, and if it is "NO", it will shift to step S211. Here, the smallest unit of the number nmax of the maximum storage of the address data after conversion is k. Moreover, by applying a limiter to the value of an address pointer P, processing in this step S209 is performed in order to make into radii the form of the locus displayed with many locus images Lo. When a limiter is not applied to the value of the above-mentioned address pointer P by this processing, the form of the locus displayed with many locus images Lo will become a circle.

[0107] At step S210, 1d of operation means adds the number k of base addresses to an address pointer P. At step S211, the value of an address pointer P is assigned to the back address AD.

[0108] At step S212, 1m of guide information management means assigns a value only with less "k" than the value of an address pointer P to the front address ad. Here, the relation between the value which the back address AD shows, and the value which the front address ad shows is as being shown below.

AD<ad [0109] At step S213, it memorizes in the area where the value of the back address AD on main memory 5 shows the address data after conversion (x y) with which 1m of guide information management means is memorized in the area which the value of the front address ad on main memory 5 shows. At step S214, 1d of operation means subtracts the number k of base addresses from the front address ad.

[0110] At step S215, 1d of operation means subtracts the number k of base addresses from the back address AD. At step S216, it judges whether it is below the value that the addition result of having added the number k of base addresses to start-address Pstart shows, if the value which the front address ad shows is "YES", it will shift to step S217, and if 1f of decision means is "NO", they will shift to step S213 again.

[0111] In a main memory 5 top, the processing to the above-mentioned step S209 - step S216 is processing which moves the address data after conversion corresponding to the tail of an arrow head (x y) to the area corresponding to the address of a big value one by one, and, so to speak, is processing as a shift register. An address pointer P is substituted for the back address AD in step S211, and only k substitutes little data for the front address ad in step S213 for memorizing in the area where the value of the back address AD shows the address data after conversion (x y) memorized in the area which the value of the front address ad shows in step S212 from an address pointer P. That is, so to speak, this processing is processing as a shift register. Moreover, in step S214, the number k of base addresses is subtracted from the

back address AD. In step S215, the number k of base addresses is subtracted from the front address ad. That the value which the front address ad shows in step S216 has judged whether it is below the value that the addition result of having added the number k of base addresses to start-address Pstart shows It is for judging whether there is any area which memorizes the address data after conversion (x y) memorized to the front address ad. That is, it is because there will be no storage area corresponding to the back address AD that the value of the front address ad turns into below the value that the above-mentioned addition result shows.
[0112] At step S217, 1m of guide information management means memorizes in the area where the back address AD of main memory 5 shows the address data after conversion of the top-most vertices of the flat surface corresponding to the tail of an arrow head (x y).
[0113] At step S218, 1h of variable setting means substitutes the addition result of start-address Pstart and the number k of base addresses for the 1st address AD 1. At step S219, 1h of variable setting means substitutes the addition result of start-address Pstart and double 2k of the number of base addresses for the 2nd address AD 2.
[0114] At step S220, 1m of guide information management means reads the address data after conversion on the main memory 5 which the 1st and 2nd address AD1 and AD2 shows (x y), and they supply the address data after the conversion concerned (x y) to the drawing processing processor 10 as a drawing instruction with texture address data and color palette data. The data memorized here in the area on the main memory 5 which the 1st address AD 1 shows are the after [conversion] address data equivalent to the tail of the arrow head currently displayed on this time. Moreover, the data memorized in the area on the main memory 5 which the 2nd address AD 2 shows are the after [conversion] address data equivalent to the tail of the arrow head with which only 1 unit period was before displayed rather than this time. That is, the address data after conversion equivalent to the tail of the arrow head with which only 1 unit period was before displayed from this time and this time (x y) are supplied to the drawing processing processor 10 as address data of one image. Therefore, the drawing processing processor 10 writes in the texture data of an arrow head as texture data of the locus image Lo on the display area of a buffer 11 based on the address data after [of above-mentioned four] conversion (x y). The color of the locus image Lo at this time is the same as the color of the arrow head Na currently displayed at present. At step S221, 1d of operation means adds the number k of base addresses to the 1st address AD 1.
[0115] At step S222, operation means 1D adds the number k of base addresses to the 2nd address AD 2. At step S223, it judges whether it is beyond the value of an address pointer P, if the value of the 1st address AD 1 is "YES", it will escape from this throw guide display routine S200, and if 1f of decision means is "NO", they will shift to step S220 again.
[0116] The processing to the above-mentioned step S217 - step S223 is processing for supplying the address data after conversion (x y) memorized by main memory 5 to the drawing processing processor 10 as address data of 2 every area and the top-most vertices of one quadrilateral. In step S218, the addition result of start-address Pstart and the number k of base addresses is substituted for the 1st address AD 1. Having substituted the addition result of start-address Pstart and double 2k of the number of base addresses for the 2nd address AD 2 in step S219 It is for supplying the address data after conversion equivalent to the tail of one arrow head (x y), and the address data after conversion equivalent to the tail of the arrow head displayed before one rather than the above-mentioned arrow head (x y) to the drawing processing processor 10 as after [conversion] address data of one image.
[0117] Moreover, in step S221, the number k of base addresses is added to the 1st address AD 1. In step S222, the number k of base addresses is added to the 2nd address AD 2. That the value which the 1st address AD 1 shows in step S223 has judged whether it is more than the address pointer P It is for judging whether there are any address data after conversion (x y) corresponding to the 2nd address AD 2 used as the address data after conversion (x y)

memorized to the 1st address AD 1 and a pair. That is, it is because there will be no address data after conversion corresponding to the 2nd address AD 2 that the value which the 1st address AD 1 shows becomes beyond the value which an address pointer P shows.
[0118] [Effectiveness in the gestalt of operation] As explained above, it sets in this gestalt. While rotating a throw object at the rate according to actuation of a controller 22 Since the image of an arrow head and a locus was displayed on real time by actuation of a controller 22 based on the direction of a throw in the throw game which was made to carry out the throw of the throw-ed object to a throw object A game player is real time, can recognize the direction of a throw of a throw-ed object, and can make the throw of the throw-ed object carry out in the direction which he meant. Furthermore, since it was made to make a sequential change of the color of an arrow head at warm color from cold color as the value of the rotational frequency of a throw object became large, a game player can recognize the remaining rotational frequencies, therefore, is real time and can recognize the throw timing of a throw-ed object.
[0119] [Modification 1] In the gestalt of the above-mentioned implementation, although the case where adjustable [of the color of an arrow head] was carried out according to the engine speed of a throw object was explained, based on the accumulation value of Data Ah, it may be made to carry out adjustable [of the color of an arrow head] whenever [throw azimuth]. That is, it is made to classify an arrow head by color more finely. Also in this case, the color of an arrow head is made into warm color from cold color as the value of the above-mentioned accumulation value turns into a large value from a small value. If it is made above, since the color number will increase, it is effective in the ability to report throw timing to a game player more finely.
[0120] [Modification 2] In the gestalt of the above-mentioned implementation, although [the color of a locus image] it is made the same color as the color of an arrow head, it is good also as white, for example. If it is made above, a game player can distinguish the locus of an arrow head and an arrow head visually, and is effective in the ability to grasp the direction of a throw certainly.
[0121] [Modification 3] In the gestalt of the above-mentioned implementation, although the case where the color of an arrow head was changed according to a rotational frequency was explained, it may be made to make the color of an arrow head into warm color from cold color as the value of throw energy becomes large. If it is made above, the timing to which the flight distance of a throw-ed object is extended can be intelligibly guided to a game player.
[0122]
[Effect of the Invention] According to above-mentioned **** this invention, the image information which shows a throw object, and the image in which the throw-ed object by which a throw is visually carried out with the throw object concerned is shown are displayed on the screen of a display means at least. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object visually, while being operated based on actuation of an actuation means In the throw game by which the throw of the above-mentioned throw-ed object is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means based on actuation of the above-mentioned actuation means Since the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown was displayed, there is effectiveness of the ability to make the direction which carries out the throw of the throw-ed object recognize to a game player.
[0123] Moreover, in the above-mentioned invention, actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object is the actuation or abbreviation rotation actuation describing an arc. Therefore, a game player can recognize whether the throw of the throw-ed object is carried out in the meant direction, if the throw of the

throw-ed object is carried out to a throw object to which timing even if a throw object is a time of performing the actuation or rotation actuation describing an arc.
[0124] Moreover, in the above-mentioned invention, a sequential indication of the above-mentioned throw guide image is given so that the orbit according to actuation of the above-mentioned throw object may be drawn. Therefore, the orbit by actuation of a throw object can be recognized only by seeing a throw guide image, therefore, it concentrates on referring to a throw guide image, and a game player is effective in the ability to extend the distance-ed of a throw-ed object more.
[0125] Moreover, in the above-mentioned invention, a sequential indication of the locus image in which the locus of the above-mentioned throw guide image is shown is given so that the orbit according to actuation of the above-mentioned throw object may be drawn. Therefore, a throw guide image can make a game player predict in which location on the screen it appears, and it is effective in the ability for it to be [the throw timing which he meant] grip-easy to a game player, and make it it by this.
[0126] Moreover, in the above-mentioned invention, the above-mentioned locus image is generated based on the coordinate information on one throw guide image, and the coordinate information on the throw guide image displayed before one rather than the throw guide image concerned. Therefore, while being able to express the exact orbit according to the location of the throw guide image displayed, it is effective in the ability to raise processing speed by reusing the coordinate information on a throw guide image.
[0127] Moreover, in the above-mentioned invention, the false three-dimensional display of the above-mentioned throw object and the above-mentioned throw guide image is carried out on the screen of the above-mentioned display means. Therefore, while taking visual matching with a throw object and a throw guide image, it is effective in the ability to give the environment which is easy to operate it to a game player.
[0128] Moreover, in the above-mentioned invention, the configuration of the above-mentioned throw guide image is a configuration which is similar to an arrow-head configuration or this at least and which has the visual direction directions function. Therefore, to a game player, the direction of a throw can be shown more clearly and it is effective in the ability to give by this the environment which is easier to operate it to a game player.
[0129] Moreover, in the above-mentioned invention, adjustable [of the color of a throw guide image] is carried out according to the rotational frequency of the above-mentioned throw object. Therefore, a game player can recognize the remaining rotational frequencies and is effective in the ability of the timing which therefore performs a throw to be shown intelligibly.
[0130] Moreover, in the above-mentioned invention, the color of the above-mentioned throw guide image changes from cold color to warm color according to the value of the rotational frequency of a throw object changing from a small value to a large value. Therefore, it is effective in the ability to give the environment which is easy to operate it by the game player.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

TECHNICAL FIELD

[Field of the Invention] This invention is applied to the game system using an optical disk, a magnetic disk, a cassette type record medium using semiconductor memory, etc. with which game data were recorded, and relates to the throw guide method of presentation in a suitable throw game, a game system, and a record medium.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

PRIOR ART

[Description of the Prior Art] Many game systems are proposed. It is the system which consists of the system which consists of a special-purpose machine and a television monitor for home use, a business-use special-purpose machine, a personal computer or a workstation, a display, and a voice output machine. A controller for a player to operate each of these systems, The record medium with which the game data which consist of game program data and data, such as an image and voice, were recorded, It consists of CPU which performs control for generation of voice or an image based on game program data, the processor for processing an image, a processor for processing voice, CRT for displaying an image, and a loudspeaker for outputting voice. As the above-mentioned record medium, there are many cassettes which contained CD-ROM, semiconductor memory, and semiconductor memory. The game structure of a system is as above.
[0003] on the other hand -- the class of game -- a way of an increment -- following -- moreover, the contents of the game -- day by day -- complexity -- and it is diversified. Recently, it is proposed by operating a controller and moving the player on a screen to what performs a sport in false on the game space formed on the screen of a television monitor. As a sport game, team event games, such as soccer and baseball, and an individual event game are proposed. As an individual event game, when it divides roughly, there are "a game which runs", "a game which swims", "a game to which the player itself flies", "a game which lifts an object", "a game which fights", "a game which applies an object on a target", and "a game whose player itself throws an object (throw)." In addition, as a throwing event, the "shot put", the "hammer throw", the "discus throw", the "javelin throw", etc. occur. When these individual events are realized as a TV game, the gestalt can predict that it is as follows. That is, the control section of a game system makes the player in the game space formed on the screen of a television monitor etc. contest by moving visually based on the contents of actuation, and an actuation condition, when a game player operates a controller.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

EFFECT OF THE INVENTION

[Effectiveness in the gestalt of operation] As explained above, it sets in this gestalt. While rotating a throw object at the rate according to actuation of a controller 22 Since the image of an arrow head and a locus was displayed on real time by actuation of a controller 22 based on the direction of a throw in the throw game which was made to carry out the throw of the throw-ed object to a throw object A game player is real time, can recognize the direction of a throw of a throw-ed object, and can make the throw of the throw-ed object carry out in the direction which he meant. Furthermore, since it was made to make a sequential change of the color of an arrow head at warm color from cold color as the value of the rotational frequency of a throw object became large, a game player can recognize the remaining rotational frequencies, therefore, is real time and can recognize the throw timing of a throw-ed object.
[0119] [Modification 1] In the gestalt of the above-mentioned implementation, although the case where adjustable [of the color of an arrow head] was carried out according to the engine speed of a throw object was explained, based on the accumulation value of Data Ah, it may be made to carry out adjustable [of the color of an arrow head] whenever [throw azimuth]. That is, it is made to classify an arrow head by color more finely. Also in this case, the color of an arrow head is made into warm color from cold color as the value of the above-mentioned accumulation value turns into a large value from a small value. If it is made above, since the color number will increase, it is effective in the ability to report throw timing to a game player more finely.
[0120] [Modification 2] In the gestalt of the above-mentioned implementation, although [the color of a locus image] it is made the same color as the color of an arrow head, it is good also as white, for example. If it is made above, a game player can distinguish the locus of an arrow head and an arrow head visually, and is effective in the ability to grasp the direction of a throw certainly.
[0121] [Modification 3] In the gestalt of the above-mentioned implementation, although the case where the color of an arrow head was changed according to a rotational frequency was explained, it may be made to make the color of an arrow head into warm color from cold color as the value of throw energy becomes large. If it is made above, the timing to which the flight distance of a throw-ed object is extended can be intelligibly guided to a game player.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

TECHNICAL PROBLEM

[Problem(s) to be Solved by the Invention] Among the above-mentioned throwing events, the shot put is performed by throwing a ball, after a player rotates 180 abbreviation. The discus throw and the hammer throw are performed by throwing a disk and a hammer, after a player does multiple-times rotation into game area. In order to realize such a throwing event as a game In the processing which sets up a player's throw energy by a certain actuation of a controller at least, and a screen top A player's image with half-rotation (shot put) or the processing displayed that more than one are rotating, half-rotation or while displaying that more than one are rotating The processing displayed that the player in game space does the throw of the throw-ed object by a certain actuation of a controller and the processing displayed that a throw-ed object flies in game space based on the above-mentioned throw energy, the direction of a throw, and an include angle are required. It is for bringing close the game held in game space by real game more visually.
[0005] However, if the direction to which a throw-ed object flies is decided with the posture of the player in game space, it will become difficult [a game player] for the player in game space to judge the timing which carries out the throw of the throw-ed object. It is in the condition of the player image by which it is indicated by sequential on a screen, and is because the player in game space has to judge the timing which carries out the throw of the throw-ed object.
[0006] Moreover, the prediction vector which is shown with the image of the arm of the player in game space in the case of the hammer throw differs from the prediction vector shown with the image of the hammer currently brandished by this player. For example, when the dynamic image of the player who does the throw of the throw-ed object is created using the technique of motion capture, inertia is expressed like a genuine article, even if visual. That is, the image of a hammer serves as this side from the image of a revolving player's arm in a hand of cut. Therefore, a game player grip-comes increasingly to be hard of the timing which makes the player in game space detaching a hammer.
[0007] Furthermore, a game which is made to carry out multiple-times rotation of the throw object, and carries out the throw of the throw-ed object like the hammer throw or the discus throw has a rotational frequency in use. Therefore, also in a game, when the rotational frequency whose rotational frequency of a throw object is in use is exceeded and the throw of the throw-ed object is not yet carried out, it is necessary to make it failure etc. It is also for securing game nature. However, counting looking at the condition that a throw object rotates what [after] rotation throw object I may rotate has a possibility of causing the fall of concentration to the game of a game player, as a result affecting the result of a game while a game player counts and causing a mistake.
[0008] This invention was made in consideration of such a point, and aims at guiding intelligibly the direction of a throw of a throw-ed object, and the remaining rotational frequencies to a game player.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

MEANS

[Means for Solving the Problem] The image which one with main this invention shows the throw-ed object by which a throw is visually carried out to the image information which shows a throw object at least with the throw object concerned is displayed on the screen of a display means. In order that the above-mentioned throw object may carry out the throw of the above-mentioned throw-ed object visually, while being operated based on actuation of an actuation means Based on actuation of the above-mentioned actuation means, the throw of the above-mentioned throw-ed object is visually carried out with the above-mentioned throw object on the screen of the above-mentioned display means. It is the throw guide method of presentation in a throw game, and the throw guide image in which the direction of a throw of the above-mentioned throw-ed object which changes with actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object at any time is shown is displayed.
[0010] Moreover, other one [main] of this inventions is the actuation or abbreviation rotation actuation on which actuation for the above-mentioned throw object to carry out the throw of the above-mentioned throw-ed object draws an arc in the above-mentioned invention.
[0011] Moreover, in the above-mentioned invention, a sequential indication of other one [main] of this inventions is given so that the above-mentioned throw guide image may draw the orbit according to actuation of the above-mentioned throw object.
[0012] Moreover, in the above-mentioned invention, a sequential indication of other one [main] of this inventions is given so that the locus image in which the locus of the above-mentioned throw guide image is shown may draw the orbit according to actuation of the above-mentioned throw object.
[0013] Moreover, other one [main] of this inventions is generated in the above-mentioned invention based on the coordinate information on a throw guide image that the above-mentioned locus image was displayed as the coordinate information on one throw guide image before one rather than the throw guide image concerned.
[0014] Moreover, in the above-mentioned invention, as for other one [main] of this inventions, the false three-dimensional display of the above-mentioned throw guide image is carried out on the screen of the above-mentioned display means.
[0015] Moreover, other one [main] of this inventions is a configuration which has the visual direction directions function in which the configuration of the above-mentioned throw guide image is similar to an arrow-head configuration or this at least in the above-mentioned invention.
[0016] Moreover, in the above-mentioned invention, as for other one [main] of this inventions, adjustable [of the color of the above-mentioned throw guide image] is carried out at least according to the rotational frequency of the above-mentioned throw object.
[0017] Moreover, the color of the above-mentioned throw guide image changes from cold color to warm color according to being changed into the value from a value with the value of the

rotational frequency of the above-mentioned throw object of the above-mentioned throw object small in the above-mentioned invention with other one [main / large] of this inventions.
[0018]
[Embodiment of the Invention] Below, sequential reference of drawing 1 - drawing 13 is carried out, and the gestalt of operation of this invention is explained at a detail.
[0019] Explanation of the gestalt of operation of this invention indicates the item explanation shown below at the head of each item, and explains it in the sequence shown below about each item.
[0020] A. Game structure of a system (drawing 1)
B. The function which CPU1 shown in drawing 1 has (drawing 2)
C. The display of an arrow head and a locus (they are drawing 4 C, D, G, and H to the drawing 3 list)
D. The example of image display (drawing 5 and drawing 6)
E. Control action by the main routine (drawing 7 - drawing 10)
F. Control action by the polygon image display routine S100 (drawing 11)
G. Control action by the throw guide display routine S200 (drawing 12 - drawing 14)
[0021] A. Game structure of a system (drawing 1)
[0022] Drawing 1 is the block diagram showing the example of the game structure of a system as a gestalt of 1 operation of this invention.
[0023] [Connection and configuration] The game system shown in this drawing 1 consists of a body of a game machine, and a record medium 30 with which the game data which become an image, voice, and a list from program data were recorded. The bus 2 by which the body of a game machine is set to CPU1 and this CPU1 from the address, data, and a control bus is connected. Into this bus 2 The graphics data generation processor 3, an interface circuitry 4, main memory 5, ROM6, the elongation circuit 7, a parallel port 8, a serial port 9, the drawing processing processor 10 and a buffer 11, the speech processing processor 13 And a buffer 14, a decoder 17 and a buffer 18, an interface circuitry 20, and memory 21 are connected, respectively. Furthermore, a television monitor 12 is connected to the drawing processing processor 10. A loudspeaker 16 is connected to the speech processing processor 13 through an amplifying circuit 15, the record-medium driver 19 is connected to a decoder 17, a controller 22 is connected to an interface circuitry 20, and it is constituted.
[0024] Here, as for the above-mentioned game system, the gestalten differ according to an application. That is, when the above-mentioned game system is constituted as home use, a television monitor 12 and a loudspeaker 16 serve as a body of a game machine with another object. Moreover, when the above-mentioned game system is constituted as business use, all the components shown in drawing 1 are contained by one housing used as one apparatus. Moreover, when the above-mentioned game system is constituted as a nucleus, it sets a personal computer and a workstation. The above-mentioned television monitor 12 corresponds to the display for the above-mentioned computers. The above-mentioned drawing processing processor 10, the speech processing processor 13, and the elongation circuit 7 It corresponds to the hardware on the add-in board carried in some game program data currently recorded on the above-mentioned record medium 30, respectively, or the expansion slot of a computer. The above-mentioned interface circuitry 4, the above-mentioned parallel port 8, the above-mentioned serial port 9, and an interface circuitry 20 correspond to the hardware on the add-in board carried in the expansion slot of a computer. Moreover, the above-mentioned buffers 11, 14, and 18 correspond to each area of main memory 5 or the expanded memory which is not illustrated, respectively. With this gestalt, the above-mentioned game system explains taking the case of the case where it is constituted as home use.
[0025] Next, each component shown in drawing 1 is explained more to a detail. The graphics data generation processor 3 will play a role of a co-processor, if CPU1 says. That is, this

graphics data generation processor 3 performs coordinate transformation and light source count, for example, the matrix of a fixed-point format and the operation of a vector, by parallel processing. The main processings of this graphics data generation processor 3 are coordinate transformation processing and light source computation. Coordinate transformation processing is processing which is due to movement magnitude data and rotation data in two-dimensional [of the image data supplied from CPU1], or the absolute-coordinate data of each top-most vertices within a three-dimension side, asks for the address on the display area of a processing-object image, and returns the address data concerned to CPU1 again. This coordinate transformation processing is explained in full detail behind.

[0026] Moreover, light source computation is processing which calculates the brightness of an image according to the vector data of a beam of light, the normal data showing the sense of the field of a polygon, and the data in which the color of a field is shown.

[0027] The above-mentioned interface circuitries 4 are objects for an interface, such as pointing devices, such as a peripheral device, for example, a mouse, and a trackball. The program data as an operating system of a game system are memorized by the above ROM 6. If it says with a personal computer, it is equivalent to BIOS (Basic Input Output System).

[0028] the intra based on MPEG (Moving Picture Engineering Group) or JPEG (Joint PictureEngineering Group) in the above-mentioned elongation circuit 7 -- elongation processing is performed to the compression image compressed by coding. elongation processing -- decoding (decoding of the data encoded by VLC:Variavle Length Code), reverse quantization processing, IDCT (Inverse Discrete Cosine Transform) processing, and intra -- it is restoration processing of an image etc.

[0029] The drawing processing processor 10 performs drawing processing to a buffer 11 based on the drawing instruction which CPU1 publishes. A buffer 11 consists of display area and non-display area. Display area is the expansion area of the data displayed on the screen of a television monitor 12. Non-display area is storage areas, such as texture data and color palette data. Here, texture data are two-dimensional image data. Color palette data are data for specifying colors, such as texture data. These data are divided and read from a record medium 30 to multiple times by CPU1 according to the advance situation of 1 time or a game, and are memorized beforehand in the non-display area of a buffer 11.

[0030] There is a drawing instruction for drawing the drawing instruction for drawing a three-dimensional image as a drawing instruction using the drawing instruction for drawing Rhine, for example and a polygon and the usual two-dimensional image. Here, a polygon is a polygonal two-dimensional image and a triangle or a square is used in this gestalt.

[0031] The drawing instruction for drawing Rhine consists of data which mean drawing initiation and the ending address of Rhine, a color, and Rhine drawing. This Rhine drawing instruction is directly published to the drawing processing processor 10 by CPU1.

[0032] The drawing instruction for drawing a three-dimensional image using a polygon consists of the polygon top-most-vertices address data on the display area of a buffer 11, texture address data in which the storage location on the buffer 11 of the texture data stuck on a polygon is shown, and brightness data which show the brightness of a texture to the color palette address-data list which shows the storage location on the buffer 11 of the color palette data in which the color of texture data is shown. Polygon top-most-vertices address data are obtained among these data by calculating based on the data to which the graphics data generation processor 3 indicates a motion of a polygon to be polygon absolute-coordinate data from CPU1, and the data in which a motion of a view location is shown. Here, it explains that polygon top-most-vertices address data are called for how.

[0033] A motion of the body on the screen of a television monitor 12 is decided by the motion of a view location to a motion of objective the very thing and this body. For example, a motion of the body on the thing to which only a body moves and the view location is being fixed, then

the screen of a television monitor 12 is a motion of the body itself. On the contrary, the motion on the thing from which there is no motion in a body and only the view location was moved, then the screen of a television monitor 12 is a motion of the view location itself. In addition, it will be easier to understand that a "view location" reads it as a "camera location." That is, a display which picturized the body is performed, moving a camera on the screen of a television monitor 12. Although the case where either a body or a view location moved was explained in order to simplify explanation, usually it is processed as both the body and the view location are moving, and the result is displayed.

[0034] Here, "a motion" of the above-mentioned body consists of a "rotation" and "movement magnitude." The rotation of the body to a view location is generated by the objective angle of rotation and the angle of rotation of a view location. Here, a rotation and an angle of rotation are expressed in procession of 3x3 by the processing for which 2x2 and three-dimension system of coordinates are used in the processing for which two-dimensional system of coordinates are used. Moreover, the movement magnitude of the body to a view location is generated by the objective location (coordinate value), the location (coordinate value) of a view location, and the angle of rotation of a view location. Here, an angle of rotation is expressed in procession of 3x3 like **** by the processing for which 2x2 and three-dimension system of coordinates are used in the processing for which two-dimensional system of coordinates are used. In addition, the objective angle of rotation based on actuation of a controller 22 and the angle of rotation of a view location are held as a table, respectively. Based on actuation of a controller 22, from the above-mentioned table, CPU1 reads the angle of rotation of a corresponding body or a view location, and it is used for it in order to calculate the rotation of a body [as opposed to a view location for the read angle of rotation], and movement magnitude.

[0035] The polygon top-most-vertices address data on display area are called for as follows so that the above explanation may show. That is, according to actuation of a controller 22, an objective angle of rotation and a location, and a list are asked for the angle of rotation of a view location, and a location by CPU1. Next, based on an objective angle of rotation and the angle of rotation of a view location, the rotation of the body to a view location is calculated by CPU1. And based on an objective location, the location of a view location, and an angle of rotation, the movement magnitude of the body to a view location is calculated by CPU1. The rotation and movement magnitude data of these bodies are expressed with the matrix of 3x3 when three-dimension system of coordinates are used and processed, as already explained.

[0036] The rotation and movement magnitude data of the above-mentioned body are given to the graphics data generation processor 3 with the absolute-coordinate data of a polygon. The graphical data generation processor 3 changes the absolute-coordinate data of a polygon into polygon top-most-vertices address data based on the rotation and movement magnitude data of the above-mentioned body. The above is processing until polygon top-most-vertices address data are obtained.

[0037] The above-mentioned polygon top-most-vertices address data show the address on the display area of a buffer 11. The drawing processing processor 10 sets up the range of the triangle shown with 3 or four polygon top-most-vertices address data, or a square 1 on the display area of a buffer 11, and writes in the texture data corresponding to the range concerned. Generally this processing is called "A texture should stick" etc. On the screen of a television monitor 12, the body with which texture data were stuck on many polygons is displayed by this.

[0038] The drawing instruction for drawing the usual two-dimensional image consists of top-most-vertices address data, texture address data, and brightness data that show the brightness of a texture to a color palette address-data list. Top-most-vertices address data are coordinate data with which the graphics data generation processor 3 carries out coordinate

transformation of the top-most-vertices coordinate data on the flat surface from CPU1, and obtains it based on the movement magnitude data from CPU1 among these data. Hereafter, it simplifies like "publishing a drawing instruction" and drawing processing is indicated.
[0039] The speech processing processor 13 makes a sound source the ADPCM data which memorized the ADPCM data read from the record medium 30 to the buffer 14, and were memorized by this buffer 14. And the speech processing processor 13 reads ADPCM data with a clock with a frequency of 44.1kHz. and -- and the speech processing processor 13 processes conversion of a pitch, addition of a noise, setup of an envelope, setup of level, addition of RIBABU, etc. to the ADPCM data read from the buffer 14. When the voice data read from a record medium 30 is PCM data, this PCM data is changed into ADPCM data by CPU1. Moreover, processing by the program data to PCM data is directly performed on main memory 5. By being processed on main memory 5, further, the voice data encoded by the data of an ADPCM format is outputted from a loudspeaker 16 as voice, after various processings which the speech processing processor 13 was supplied and were mentioned above are performed.
[0040] The record-medium drivers 19 are a hard disk drive, an optical disk drive, a flexible disk drive, a silicon disk drive, a cassette medium readout machine, etc. Record media 30 are a hard disk, an optical disk, a flexible disk, semiconductor memory, etc. From a record medium 30, the record-medium driver 19 reads an image, voice, and game program data, and supplies the read data to a decoder 17. A decoder 17 supplies the data which performed error correction processing by ECC (Error Correction Code), and performed error correction processing to main memory 5 or the speech processing processor 13 to the playback data from the record-medium driver 19.
[0041] Memory 21 consists of memory of a holder and a card mold. The memory of a card mold is for holding the various parameters of a game like holding the condition for example, at the termination time. A controller 22 consists of cross-joint key [which consists of the left key L, the right key R, an upper key U, and a bottom key D], left carbon button 22L and right carbon button 22R, start button 22a, select button 22b, 1st carbon button 22c, 2nd carbon button 22d, 3rd carbon button 22e, and 4th carbon button 22f. A cross-joint key gives the command with which a game player shows four directions to CPU1. Start button 22a is for a game player to direct initiation of the game program data loaded from a record medium 30 to CPU1. Select button 22b is for a game player to direct the various selections about the game program data loaded to main memory 5 from a record medium 30 to CPU1. In addition, left key 22L, right key 22R, and the 1st - carbon button [4th /c / 22 /,d / 22 /,e / 22 /, and 22f] function change with game program data loaded from a record medium 30.
[0042] [Actuation] An electric power switch (not shown) is turned ON and a power source is supplied to a game system. If the record-medium driver 19 is loaded with the record medium 30 at this time, CPU1 directs read-out of the game data from a record medium 30 to the record-medium driver 19 based on the operating system memorized by ROM6. Thereby, the record-medium driver 19 reads an image, voice, and game program data from a record medium 30. The image, voice, and game program data which were read are supplied to a decoder 17, and error correction processing is performed here. The image data to which error correction processing was performed in the decoder 17 is supplied to the elongation circuit 7 through a bus 2, after elongation processing mentioned above is performed here, is supplied to the drawing processing processor 10, and is written in the non-display area of a buffer 11 by this drawing processing processor 10.
[0043] The voice data with which error correction processing was performed in the decoder 17 is supplied to main memory 5 or the speech processing processor 13, and is written in main memory 5 or a buffer 14. Moreover, the game program data with which error correction processing was performed in the decoder 17 are supplied to main memory 5, and are written in

this main memory 5. Henceforth, CPU1 advances a game based on the contents which a game player directs through a controller 22 in the game program data memorized by main memory 5 and a list. That is, CPU1 performs control of an image processing, control of speech processing, and control of internal processing suitably based on the contents of directions directed from a game player through a controller 22. Control of an image processing is supply to the graphics data generation processor 3 of the rotation and movement magnitude data which were mentioned above, or absolute-coordinate data, issue of the drawing instruction containing address data and brightness data on the display area of the buffer 11 for which the graphics data generation processor 3 asked, etc. Control of speech processing is assignment of issue of the voice output command to the speech processing processor 13, level, RIBABU, etc. Control of internal processing is an operation according to actuation of a controller 22 etc.
[0044] B. The function which CPU1 shown in drawing 1 has (drawing 2)
[0045] Drawing 2 is the explanatory view showing the function which CPU1 shown in drawing 1 has. CPU1 is read from the record medium 30 shown in drawing 1 , and has the function shown in drawing 2 by reading the program data memorized by main memory 5. The function of CPU 1 shown in this drawing 2 consists of information setting means 1e, 1f of decision means, 1g of drawing instruction issue means, 1h of variable-setting means, number acquisition means of coma 1i, migration and rotation acquisition means 1j, polygon information-management means 1k, and 1 m of guide information-management means as a result of button-grabbing detection means 1a, view location data setting means 1b, display-rectangle information extract means 1c, and 1 d of operation means. These means serve as a subject of control who explains in Item E - Item G, respectively.
[0046] C. The display of an arrow head and a locus (they are drawing 4 C, D, G, and H to the drawing 3 list)
[0047] Drawing 3 A for example, the explanatory view for explaining the method of a display of the throw guide which consists of the arrow head and locus on the screen which consists of 480 pixels long and 640 pixels wide and drawing 3 B The explanatory view showing the absolute coordinate of the image of an arrow head and drawing 3 C are the explanatory views for explaining notionally the image deformation processing when setting up an arrow head on a three-dimension coordinate based on an absolute coordinate, and the movement magnitude and rotation data of the arrow head shown in drawing 3 B. Moreover, the explanatory view in which drawing 4 C shows an example of an arrow-head absolute coordinate and drawing 4 D The arrow-head absolute-coordinate data shown in drawing 4 C are based on movement magnitude and rotation data. Explanatory view and drawing 4 G which shows an example of the arrow-head address after conversion changed by the graphics data generation processor 3 shown in drawing 1 The explanatory view showing an example of the arrow-head posterior part address after conversion which is equivalent to the tail of an arrow head among the arrow-head addresses after conversion shown in drawing 4 D, and drawing 4 H are the explanatory views showing an example of the address for locus generation obtained from the arrow-head posterior part address after conversion shown in drawing 4 G.
[0048] Let it be the purpose to guide the remaining rotational frequencies of a throw object to the timing to which the throw object in game space used as the direction of a throw which a game player means carries out the throw of the throw-ed object, and a list intelligibly for a game player in this gestalt. This purpose is attained by displaying the locus Lo of the arrow head Na which shows the direction where a throw object carries out the throw of the throw-ed object, and this arrow head Na on the screen of a television monitor 12, as shown in drawing 4 A. And the direction which an arrow head shows is in agreement in the direction of a throw of the throw-ed object which the posture of throw objects, such as a player displayed with the arrow head, shows. That is, the arrow head which shows the direction of a throw according to the posture of the throw object displayed every moment on the screen is displayed every

moment with a throw object.
[0049] Moreover, an arrow head Na is displayed in order toward the direction shown by Orbit OR top and the arrow head. At this time, two address data a1 and a2 of the part of the tail of the arrow head Na by which it is indicated by current, and two address data b2 and b1 of the part of the tail of the arrow head Nb displayed before one are used, and one image is generated and displayed. This image is the locus image Lo used as a locus of an arrow head Na. As shown in this drawing 3 A, the locus image Lo is constituted by two triangles formed by tying the address data a1 of the part of the tail of an arrow head Na, and the address data b2 of the part of the tail of an arrow head Nb, and tying the address data a2 of the part of the tail of an arrow head Na, and the address data b1 of the part of the tail of an arrow head Nb.
[0050] And the rotational frequency of a throw object is further expressed by the color of an arrow head Na. For example, the color of the arrow head Na when there are few values which show a rotational frequency is set as purple, the color of an arrow head Na is set as warm color one by one as the value which shows a rotational frequency becomes large, and the color of the arrow head Na when the value which shows a rotational frequency is the largest is set as red. As for an arrow head, in the case of the shot put, a throw game is displayed in red from the beginning. It is because rotational frequencies in use are 0.5 rotations in a shot put game. Moreover, as for an arrow head, in the case of the discus throw, a throw game is displayed [yellow and 2 rotation eye] in red by 1 rotation eye, respectively. It is because rotational frequencies in use are two rotations in a discus throw game. moreover, a throw game -- the case of the hammer throw -- an arrow head -- in 1 rotation eye, it is displayed [blue and 3 rotation eye / green and 4 rotation eye / yellow and 5 rotation eye] in red by purple and 2 rotation eye, respectively. Although rotational frequencies in use are four rotations in a hammer throw game, it is because it is considering as five rotations with this gestalt. In addition, on processing, based on the division value (below decimal point cut-off) acquired by 1d of operation means doing the division of the data Ah by 360 whenever [throw azimuth], the color of an arrow head is called for, when 1g of drawing instruction issue means refers to a table. A table consists of data in which the color palette address data or the color corresponding to many division values and the division value of these large number is shown, respectively.
[0051] Moreover, in this gestalt, when the maximum above-mentioned rotational frequency is exceeded, it considers as failure. This mistake is made in the foul in a real game.
[0052] The above throw guide displays are expressed as real time according to the posture of a throw object. Therefore, a game player can know the direction of a throw and the remaining rotational frequency when a throw object carries out the throw of the throw-ed object at present. Therefore, by the above-mentioned throw guide display, a game player can recognize the direction of a throw and the timing to mean of the throw-ed object which he means, can operate a controller 22 based on this recognition, and can carry out the throw of the throw-ed object to a throw object. That is, the throw game excellent in the user interface is realizable. Hereafter, the processing for carrying out the above-mentioned guide display to a detail is explained more.
[0053] An arrow head is a configuration as shown for example, in drawing 3 A. And the absolute coordinate of this arrow head is the coordinate data of the angle of the quadrilateral surrounding the arrow head on the three-dimension flat surface containing the arrow head at the time of setting a core to "0", as shown in drawing 3 A. Moreover, this arrow-head absolute-coordinate data is data loaded to main memory 5 from the record medium 30. In the example of the arrow-head absolute-coordinate data shown in this drawing 3 B and drawing 4 B (x1, y1, z1) -- x2, and (y2, z2) -8, (0, 10), and (x3, y3, z3) -- (8, 0, -10), and (x4, y4, z4) -- (- it is 8, 0, and -10). [(8, 0 10), and]
[0054] Drawing 3 C shows notionally transform processing by the graphics data generation

processor 3. If the arrow-head absolute-coordinate data shown in drawing 3 R>3B, the movement magnitude mentioned above, and rotation data are given from CPU1, the graphics data generation processor 3 will set up an arrow head on a three-dimension coordinate, as shown in drawing 3 C. The location and angle of rotation of a camera shall be fixed to the location of explanation a throw object and a throw-ed object always appear for convenience.
[0055] The address data of the top-most vertices of the quadrilateral surrounding the arrow head set up on the three-dimension coordinate are changed into the address data on the display area of a buffer 11. The example of this arrow-head address after conversion is shown in drawing 4 D. The arrow-head address after conversion is the address data on the display area of buffer memory 11, as shown in drawing 4 D. for example, the addresses a1, a2, a3, and a4 on the display area of the buffer 11 of the angle of the quadrilateral surrounding the arrow head Na at present shown in drawing 3 A -- respectively -- (560, 320), and (544, 324) -- it is [and / (560 394) / (594 368)]. moreover, the addresses b1, b2, b3, and b4 on the display area of the buffer 11 of the angle of the quadrilateral which surrounds the arrow head Nb in front of one from this time -- respectively -- (424, 342), and (408, 346) -- it is [and / (424 416) / (458 390)] (refer to drawing 3 A and drawing 4 D).
[0056] Next, the locus image Lo is explained. It explains as that whose arrow head which the arrow head Na was displayed at present, and the arrow head Nb was displayed before predetermined unit time amount, and was displayed in the past is only an arrow head Nb.
[0057] As shown in drawing 3 A, when the arrow head Na is displayed, the arrow-head address data after conversion of the arrow head Na currently displayed at present as shown in drawing 4 D, and the address data after conversion of the arrow head Nb currently displayed from this time at the time before one are memorized by main memory 5. On the other hand, as shown in drawing 4 G, only the address which is equivalent to the tail of an arrow head among the arrow-head addresses after conversion shown in drawing 4 D is memorized by main memory 5 as after [conversion] arrow-head posterior part address data. As it turns out that drawing 3 A is referred to, the addresses for a tail of the arrow head Nb displayed the arrow head Na currently displayed at present and one frame ago are a1, a2, and b1 and b2, respectively. And these values are (560, 320), and (544 324) (424 342) (408, 346) as shown in the lower part of this drawing 3 A. These values are memorized as the after [conversion] arrow-head posterior part address in other area of main memory 5 as shown in drawing 4 R>4G.
[0058] And as address data for locus generation, after the arrow-head posterior part address data after both conversion of arrow heads Na and Nb are memorized by main memory 5, they are supplied to the drawing processing processor 10, as shown in drawing 4 H. The drawing processing processor 10 will write in the texture data of an arrow head as locus image data on the display area of a buffer 11 based on these address data, if the drawing instruction containing the above-mentioned address data for locus generation is received. Thereby, the locus image Lo is displayed on the screen so that a slash may be attached and shown in drawing 3 A. This locus image Lo appears visual just like the locus of an arrow head Na. And the orbit OR of an arrow head is visually expressed by giving a sequential indication of an arrow head and the locus.
[0059] Since the number of the arrow heads displayed on this time and the past is two and the locus image Lo displayed also serves as a large number in this example when this number is a large number although the number of the locus images Lo displayed is one, the die length of the locus image Lo displayed on the arrow head currently displayed on the screen by following becomes long.
[0060] If the orbit OR of an arrow head is expressed by the display of the locus image Lo, since the location where an arrow head is displayed beforehand understands a game player, only in the part, its response to the direction of a throw shown by the arrow head will improve.

That is, the direction where an arrow head is displayed on the orbit OR currently expressed can predict the display position of an arrow head rather than an arrow head is displayed on the place which does not have anything one by one, and an actuation reaction when the arrow head which should carry out a throw to a throw object by this is displayed can be raised.
[0061] Next, the color of an arrow head Na is explained. As already explained, with this gestalt, the color of an arrow head Na is changed according to rotation of a throw object. This was already explained. Usually, cold color is used in order to express a thing small like the lowness of temperature. Moreover, warm color is used in order to express the large thing like the height of temperature. In this gestalt, the color of an arrow head Na is changed into warm color from cold color one by one according to a rotational frequency increasing. Therefore, a game player can recognize the remaining rotational frequencies. In other words, a game player can recognize the timing which carries out the throw of the throw-ed object to the throw object in game space. Moreover, red is usually used as a color expressing "risk", a "limitation", etc. Therefore, if the color of an arrow head is brought close to red as a rotational frequency rises, a game player can recognize certainly that the remaining rotational frequencies decrease.
[0062] D. The example of image display (drawing 5 and drawing 6)
[0063] Drawing 5 A - drawing 5 D is the explanatory view showing the example of a screen display in a hammer throw game. The player Ma in game space extracts four examples of a screen display in many a series of frames after starting the actuation for throwing Hammer Ba until it finishes throwing Hammer Ba, and shows serially this drawing 5 A - drawing 5 D. Serial sequence is drawing 5 A, drawing 5 B, drawing 5 C, and drawing 5 D. The example of a screen display at the time of drawing 5 D throwing Hammer Ba for the example of a screen display which is performing rotation actuation just before drawing 5 C throws Hammer Ba for the example of a screen display at the time of starting rotation actuation for drawing 5 B throwing Hammer Ba for the example of a screen display at the time of starting actuation for drawing 5 A throwing Hammer Ba is shown. In addition, a sign gives a sign only to drawing 5 B for convenience. Moreover, each example of a screen display is chosen from a full screen when a multiple-times game is performed.
[0064] One screen consists of a window W for displaying the image for a display as a result of a player's besides a background image Ma, the image Ba of a hammer, the image Na of an arrow head, and the locus image Lo, and a guide image Gi for an include-angle display as shown in drawing 5 B. Moreover, Lines LL and LR are displayed into a background image. The alphabetic character (this example 1 P:1 person) which shows the number of game players is displayed on the upper part of the left end part of Window W, the bar graph which shows throw energy is displayed on that lower part, and the alphabetic character (this example PLAYER1) which shows the how many persons' game player it is is further displayed on that lower part. Moreover, the alphabetic character of "1ST", "2ND", and "3RD" is displayed in the center of Window W, respectively. These are "the 1st time", "the 2nd time", and "the 3rd semantics", respectively. And the alphabetic character which shows flight distance or foul is displayed on the right-hand of the alphabetic character of the above "1ST", "2ND", and "3RD." It is shown by drawing 5 A that the flight distance in the 1st throw is [the flight distance in "59.33M" (59m 33cm is meant) and the 2nd throw] "68.39M" (68m 39cm is meant). In addition, "x" shown in the right-hand side of the alphabetic character which shows the flight distance of "1ST", and "O" shown in the right-hand side of the alphabetic character which shows the flight distance of "2ND" to a list show that the flight distance in "2ND" is an effective value rather than the flight distance in "1ST." Moreover, in drawing 5 B, it is shown by the display of "FOUL x" that the 1st throw is failure, i.e., foul.
[0065] Moreover, it is displayed that it becomes small, so that the locus image Lo is so large that it is close to an arrow head and it is far as shown in drawing 5 B.
[0066] E. Control action by the main routine (drawing 7 - drawing 10)

[0067] Drawing 6 - drawing 9 are the flow charts for explaining the control action by the main routine of a throw game. A throw game says that a throw-ed object carries out a throw into game space with the throw object in game space according to actuation of the controller 22 by the game player. As a throwing event, game, for example, the shot put, that rotation stripes carry out [a throw object] the throw of the **** throw object, the discus throw, and the hammer throw are mainly known. Therefore, in the case of a shot put game, the throw game shown in drawing 6 - drawing 9 sets. Read a "throw object" with a "player", read a "throw-ed object" with a "cannonball", change, and, in the case of a discus throw game, it sets. A "throw object" is read with a "player", a "throw-ed object" is read with a "disk", and it changes, and in the case of a hammer throw game, a "throw object" shall be read with a "player", a "throw-ed object" shall be read with a "hammer", and it shall change to it.
[0068] In addition, it is the control action by the operating system memorized by ROM6 which showed only step S1 to drawing 1 . Other steps are the control action by the game program data read from the record medium 30. Moreover, the subject of control by game program data is each means as a function of CPU1 shown in drawing 2 , as already explained.
[0069] At step S1, the record-medium driver 19 reads an image, voice, and game program data from a record medium 30 with the instruction of an operating system. Program data are memorized by main memory 5 among the read data. Thereby, CPU1 has the function shown in drawing 2 . In addition, at this time, an image, i.e., texture data, shall be memorized in the non-display area of the buffer 11 of the drawing processing processor 10, and a texture number shall be assigned, respectively. Moreover, voice data shall be memorized by the buffer 14 of the speech processing processor 13, and voice number data shall be assigned, respectively. Usually, although all images and voice data are not held in step S1 at buffers 11 and 14, all images and voice data consider as the thing of explanation loaded in step S1 for convenience. At step S2, it judges whether it is the no on which start button 22a of a controller 22 was pushed, and if button-grabbing detection means 1a is "YES", it will shift to step S3.
[0070] At step S3, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of a selection image is shown to the drawing processing processor 10 shown in drawing 1 . The drawing processing processor 10 develops the image data of a selection image on the screen of a buffer 11 based on the above-mentioned drawing instruction. Thereby, a selection image is displayed on the screen of a television monitor 12. In step S4, it judges whether start button 22a of a controller 22 was pushed, and if button-grabbing means 1a is "YES", it will shift to step S5.
[0071] CPU1 sets to the selected game at step S5. Here, with reference to the selection image with which the game player was displayed ["it is selected" and] at step S3, a game is chosen using a cross-joint key and it means pushing start button 22a next. Moreover, the character in other, for example, waging war, mold fighting game [game / itself] etc. is included with a "game" here. In short, it is the selection matter before a game is actually started. In this step S5, it considers as the thing of explanation as which the throw game was chosen for convenience. At step S6, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of the initial image of the selected game is shown to the drawing processing processor 10. Thereby, the drawing processing processor 10 writes in the image data of an initial image on the display area of a buffer 11. Thereby, an initial image is displayed on the screen of a television monitor 12.
[0072] At step S7, 1h of variable setting means resets the flag and variable which are held to main memory 5, respectively. At step S8, it judges whether 1st carbon button 22c was pushed, if it is "YES", it will shift to step S9, and if button-grabbing detection means 1a is "NO", it will shift to step S11. Actuation of 1st carbon button 22c is for controlling the rotational speed of the throw object in game space.
[0073] In step S9, 1d of operation means adds the reference speed data s to the speed data

Sd at the time of a rise.

[0074] At step S10, number acquisition means of coma 1i asks for the number data fd of coma according to the value of the speed data Sd. This number data fd of coma is called the number data of conversion coma. The number data fd of conversion coma are called for by referring to a table. A table consists of much speed data Sd and number data of coma of a large number registered about the speed data Sd of these large number, respectively, and is loaded to main memory 5 from a record medium 30. In addition, you may make it ask for the number of coma by performing a predetermined operation using the speed data Sd.

[0075] At step S11, 1d of operation means subtracts the reference speed data m from the speed data Sd at the time of descent. At step S12, if the value of the speed data Sd judges whether it is minus and is "YES", it will shift to step S13, and if 1f of decision means is "NO", they will shift to step S10.

[0076] At step S13, 1h of variable setting means substitutes "0" for the speed data Sd. When button-grabbing detection means 1a judges whether 1st carbon button 22c was pushed at the above-mentioned step S8 and it judges that 1st carbon button 22c is pushed When it is judged as that on which shift to step S9, the speed data Sd are made to increase, and 1st carbon button 22c is not pushed It is for setting up the value of the speed data Sd according to the count of press of 1st carbon button 22c per unit time amount by the game player to shift to step S11 and to decrease the speed data Sd. That is, if there are many counts by which a game player pushes 1st carbon button 22c into unit time amount, it will go up, the rotational speed, i.e., the throw energy, of a throw object in game space. On the other hand, if the count by which a game player pushes 1st carbon button 22c into unit time amount is low, it will descend, the rotational speed, i.e., the throw energy, of a throw object in game space. Although not shown as a step in the flow chart, the display of the above-mentioned throw energy is expressed by change of a bar graph shown in drawing 5 A - drawing 5 D.

[0077] At step S14, 1d of operation means adds the number data fd of conversion coma to the number data FD of coma. At step S15, number acquisition means of coma 1i reads Data Ah from the table TBL currently held at main memory 5 whenever [corresponding to the number data FD of coma called for in step S14 / throw azimuth]. Here, Table TBL consists of data Ah whenever [throw azimuth / of a large number registered, respectively about much number data FD of coma, and the number data FD of coma of these large number] as shown in drawing 4 A.

[0078] At step S16, 1h of variable setting means reads the movement magnitude data according to a value and rotation data of the address data on the main memory 5 of the absolute-coordinate data of the polygon of a throw object, and the number data FD of coma from a table, the above-mentioned movement magnitude data are substituted for the movement magnitude variable MO, and the above-mentioned rotation data are substituted for the rotation variable RO for the above-mentioned address data at the address variable ADD, respectively. Here, rotation data are obtained by the above-mentioned address data and the movement magnitude data list by referring to a table. This table becomes the address data of the number of coma from the minimum value to maximum, and a large number registered according to the value of these numbers of coma, respectively, and a movement magnitude data list from rotation data.

[0079] Polygon image display processing is performed at step S100. This polygon image display routine S100 is explained in full detail behind. Throw guide display processing is performed at step S200. This throw guide display routine S200 is explained in full detail behind.

[0080] At step S17, it judges whether it is below the value of the minimum value data Ahmin of whenever [throw azimuth], whenever [throw azimuth], if the value of Data Ah is "YES", it will shift to step S18, and if 1f of decision means is "NO", they will shift to step S22. Here, the

minimum value data Ahmin of whenever [throw vectorial angle] are the smallest include angle and an include angle to which a throw-ed object falls in un-effective area with allowances, as shown in drawing 4 A. Here, un-effective area is area out of range shown among Lines LL and LR as shown in drawing 4 B, drawing 5 A - drawing 5 D, respectively.

[0081] By the shot put, the maximum engine speed at the time of the throw for which it opts in this gestalt is become to "2" by "0.5" and the discus throw, and has become "5" in the hammer throw. Then, in this gestalt, the maximum of the number of coma is decided according to the above-mentioned maximum engine speed. For example, if the value of the number of the maximum coma shall be set as "240" for the rotational frequency set up by ****** by 4 times, the value of the number of the maximum coma in the game whose rotational frequency is 2 times will be set to "120." Moreover, in the above-mentioned track and field, "failure" means "foul."

[0082] In step S17, when 1f of decision means judges it as "NO", it shifts to step S18 and result information setting means 1e supplies the alphabetic data in which "FOUL x" is shown to the drawing processing processor 10 at this step S18. Thereby, as shown in drawing 5 B, the alphabetic character which shows "FOUL x" is displayed on the central part of Window W as an image. At step S300, throw object image display processing for failure is performed. This throw object image display routine S300 for failure consists of processing performed at setting processing and step S100 which are performed at step S32 mentioned later. On the screen, a motion of the throw object at the time of failure is expressed, for example like a throw object collapsing.

[0083] At step S19, it judges whether which carbon button was pushed by the existence of the data which 1f of decision means shows the carbon button pushed from button-grabbing detection means 1a, if it is "YES", it will shift to step S20, and if it is "NO", it will shift to step S300 again. At step S20, 1d of operation means adds "1" to the count data Th of a throw.

[0084] At step S21, it judges whether the count data Th of a throw are larger than the maximum Thmax of the count data of a throw, if it is "YES", it will shift to step S3 again, and if 1f of decision means is "NO", they will shift to step S8 again. At step S22, it judges whether 2nd carbon button 22d was pushed, if it is "YES", it will shift to step S23, and if button-grabbing detection means 1a is "NO", it will shift to step S8 again. Here, throw timing is determined as the throw vertical include-angle data Av of a throw-ed object 2nd carbon button 22d. With this gestalt, while 2nd carbon button 22d is pushed, the sequential increment of the value of the throw vertical include-angle data Av is carried out. In addition, the value of the above-mentioned throw vertical include-angle data Av is displayed on real time with Guide Gi, as shown in drawing 5 A - drawing 5 D.

[0085] At step S23, 1d of operation means adds the criteria include-angle data z to the throw vertical include-angle data Av. At step S24, it judges whether it is larger than the value of the maximum data Avmax of the throw vertical include-angle data Av, if the value of the throw vertical include-angle data Av is "YES", it will shift to step S25, and if 1f of decision means is "NO", they will shift to step S26.

[0086] At step S25, 1h of variable setting means substitutes the maximum data Avmax of the throw vertical include-angle data Av for the throw vertical include-angle data Av. At step S26, it judges whether 2nd carbon button 22d was detached, if it is "YES", it will shift to step S27, and if button-grabbing detection means 1a is "NO", it will shift to step S23 again. When 2nd carbon button 22d is detached, the throw of the throw-ed object is carried out with the throw object in game space with the value of the throw vertical include-angle data Av at the time.

[0087] At step S27, 1h of variable setting means initializes the rate vector data of a throw-ed object based on Data Ah and the value of the throw vertical include-angle data Av the value of the speed data Sd, and whenever [throw azimuth]. Here, rate vector data shows the location on a three-dimension coordinate, and consists of x, and (y, z). Here, "initialization" means

setting up the rate vector data decided by the three above-mentioned values. At step S28, 1d of operation means adds gravitational acceleration vector data to the rate vector data initialized at step S27. Here, gravitational acceleration vector data is a constant for changing the location on the three-dimension coordinate which the above-mentioned rate vector data shows (x y, z).
[0088] At step S29, 1d of operation means adds rate vector data (x y, z) to the location data (x y, z) of a non-throw object. Here, the location data of a non-throw object show the location of the throw-ed object on a three-dimension coordinate, and consist of x, and (y, z). At step S30, if the height of a throw-ed object judges whether it is "0" and is "YES", it will shift to step S31, and if 1f of decision means is "NO", they will shift to step S32. Here, height means the height on the screen of the throw-ed object for every frame.
[0089] At step S31, 1h of variable setting means sets the height of a throw-ed object to "0." step S32 -- 1h of variable setting means -- the above-mentioned movement magnitude data are substituted for the movement magnitude variable MO, and the above-mentioned rotation data are substituted for the address variable ADD for the address data on the main memory 5 of the absolute-coordinate data of the polygon of a throw-ed object at the rotation variable RO, respectively. Here, rotation data are obtained by the above-mentioned movement magnitude data list by referring to a table. This table becomes the movement magnitude data list of a large number registered according to the value of the location data of the throw-ed object from the minimum value to maximum, and these location data, respectively from rotation data. Of course, based on the above-mentioned location data, you may ask for the above-mentioned movement magnitude data and rotation data by the operation at any time.
[0090] Polygon image display processing is performed at step S100. At step S400, after [a throw] throw object image display processing is performed. This throw object image display routine S400 is a routine which consists of processing performed at the processing performed at step S32, and step S100. The motion after a throw object carries out the throw of the throw-ed object on the screen is expressed.
[0091] At step S33, 1d of operation means finds flight distance. And result information setting means 1e supplies the alphabetic data in which flight distance is shown to the drawing processing processor 10. At step S34, 1g of drawing instruction issue means supplies the drawing instruction in which the display of the image for a result display is shown to the drawing processing processor 10. On the screen of a television monitor 12, the alphabetic character which shows the flight distance of a throw-ed object is displayed on the central part of the window W shown in drawing 5 A - drawing 5 D as an image by this.
[0092] At step S35, based on the existence of the data which 1f of decision means shows the manual operation button from button-grabbing detection means 1a, it judges whether which carbon button was pushed, and if it is "YES", it will shift to step S36. At step S36, 1g of drawing instruction issue means publishes the drawing instruction in which drawing of the image at the time of a success is shown to the drawing processing processor 10. Thereby, the drawing instruction processor 10 writes in the image data at the time of a success on the display area of a buffer 11. Therefore, the image at the time of a success is displayed on the screen of a television monitor 12. Here, the image at the time of a success is an image of alphabetic characters, such as "GOOD", etc., for example, it is displayed on the central part of the window W shown in drawing 5 A - drawing 5 D.
[0093] At step S500, throw object image display processing is performed at the time of a success. The throw object image display routine S500 is a routine which consists of processing performed at the processing performed at step S32, and step S100 at the time of this success. On the screen, a motion of the throw object at the time of a success is expressed like a throw object flying and being over, for example. Replay image display processing is performed at step S600. The actuation information on the controller 22 according [this replay image display processing] to a game player is used. That is, the actuation information on the

controller 22 by the game player is memorized in detail by main memory 5. And in this replay image display processing, sequential processing is performed based on the above-mentioned actuation information. Thereby, the image display condition based on the actuation which the game player performed is reproducible.

[0094] At step S37, it judges whether which carbon button was pushed by the existence of the data which 1f of decision means shows the contents of actuation from button-grabbing detection means 1a, and if it is "YES", it will shift to step S20 again.

[0095] F. Control action by the polygon image display routine S100 (drawing 11)

[0096] Drawing 10 is a flow chart for explaining the control action by the polygon image display routine S100. By this polygon image display routine S100, polygon display processing of a throw object or a throw-ed object is performed. One throw object and throw-ed object consist of many polygons. As shown in drawing 4 R>4E, the polygon is memorized by main memory 5 as absolute-coordinate data (x y, z) of much polygon top-most vertices. And these absolute-coordinate data (x y, z) are changed into the polygon address data after conversion on the two-dimensional flat surface shown in drawing 4 F (x y) by the graphics data generation processor 3 based on the movement magnitude data and rotation data which were mentioned above. The polygon address data after conversion (x y) shown in this drawing 4 F are supplied to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. The drawing processing processor 10 writes in texture data on the display area of a buffer 11 based on the above-mentioned polygon address data after conversion based on a drawing instruction. Thereby, on the screen of a television monitor 12, the throw object and throw-ed object which consist of many polygons are displayed.

[0097] At step S101, polygon information management means 1k reads the absolute-coordinate data (x y, z) of the top-most vertices of the polygon on the main memory 5 which the value of the address variable ADD shows from main memory 5. At step S102, polygon information management means 1k supplies the data of the absolute coordinate of the top-most vertices of a polygon, the movement magnitude substituted for the movement magnitude variable MO, the rotation substituted for the rotation variable RO, the vector of a beam of light, and the normal of a polygon to the graphics data generation processor 3, respectively. the graphics data generation processor 3 -- each above-mentioned data -- being based -- the polygon address data after conversion (x y) -- and it asks for brightness data, respectively and these data are supplied to polygon information management means 1k, respectively. At step S103, polygon information management means 1k writes the after [conversion] polygon address data (x y) and brightness data from the graphics data generation processor 3 in main memory 5, respectively.

[0098] At step S104, it judges whether all the absolute-coordinate data of the top-most vertices of a polygon were changed into the polygon address data after conversion, if it is "YES", it will shift to step S105, and if 1f of decision means is "NO", they will shift to step S102 again. At step S105, 1g of drawing instruction issue means reads after [conversion] address data (x y), and brightness data from main memory 5, and they supply after [the conversion concerned] address data (x y), and brightness data to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. Thereby, the drawing processing processor 10 writes in the texture data of a throw object on the display area of a buffer 11 based on the above-mentioned address data after conversion (x y). Therefore, on the screen of a television monitor 12, image data, such as a throw object which consists of many polygons, and a throw-ed object, is displayed.

[0099] At step S106, it judges whether all data were transmitted, if it is "YES", it will escape from this polygon image display routine S100, and if 1f of decision means is "NO", they will shift to step S105 again.

[0100] G. Control action by the throw guide display routine S200 (drawing 12 - drawing 14)

[0101] Drawing 11 - drawing 13 are the flow charts for explaining the control action by the throw guide display routine. A throw guide consists of an image of an arrow head, and a locus image in which the locus of this arrow head is shown.
[0102] At step S201, 1m of guide information management means reads the absolute-coordinate data of the top-most vertices of the quadrilateral surrounding an arrow head from main memory 5. At step S202, migration and rotation acquisition means 1j obtain the movement magnitude data and rotation data on the three-dimension coordinate of the quadrilateral surrounding an arrow head according to the distance data from the center of rotation O indicated to be the value of Data Ah to drawing 3 A whenever [throw azimuth]. Here, the above-mentioned distance data are a fixed value.
[0103] At step S203, 1m of guide information management means supplies the absolute-coordinate data, movement magnitude data, and rotation data of an arrow head to the graphics data generation processor 3. Thereby, the graphics data generation processor 3 changes the absolute-coordinate data of an arrow head on a three-dimension coordinate based on movement magnitude data and rotation data, from the coordinate data obtained by this conversion, obtains the address data after conversion on two-dimensional (x y), and supplies the address data after this conversion (x y) to 1m of guide information management means. At step S204, 1m of guide information management means writes the address data after conversion from the graphics data generation processor 3 (x y) in main memory 5.
[0104] At step S205, 1g of drawing instruction issue means reads the address data after conversion (x y) from main memory 5, and they supply the address data after the conversion concerned (x y) to the drawing processing processor 10 as a drawing instruction with texture address data and color palette address data. In addition, 1g of drawing instruction issue means is acquired based on the operation value to which 1d of operation means carries out the color palette address data for displaying an arrow head by the color according to an engine speed. An operation value is calculated here by doing the division of the data Ah by 360 (degree) whenever [throw azimuth]. Thereby, the drawing processing processor 10 writes in the texture data of an arrow head on the display area of a buffer 11 by the color specified by the color palette based on the address data after conversion (x y). In addition, the decision of a color is made by reference of a table. A table consists of the data or color palette address data for specifying the color registered, respectively about the division values and these multiplication values from the minimum value calculated by 1d of operation means to maximum.
[0105] At step S206, if the value of an address pointer P judges whether it is start-address Pstart and is "YES", it will shift to step S207, and if 1f of decision means is "NO", they will shift to step S209. Here, Pstart is a start address in the area where the address data after conversion are memorized. At step S207, 1d of operation means adds the number k of base addresses to an address pointer P. Here, the value of the number k of base addresses is capacity value required to memorize two address data.
[0106] At step S208, 1m of guide information management means memorizes the address data after conversion of the flat surface corresponding to the tail of an arrow head (x y) in the area which the value of the address pointer P on main memory 5 shows. And it escapes from this throw guide display routine S200. At step S209, it judges whether the value which an address pointer P shows is smaller than the value which the subtraction result of having reduced the number k of base addresses indicates to be start-address Pstart from an addition result with the number nmax of the maximum storage of the address data after conversion, if 1d of operation means is "YES", they will shift to step S210, and if it is "NO", it will shift to step S211. Here, the smallest unit of the number nmax of the maximum storage of the address data after conversion is k. Moreover, by applying a limiter to the value of an address pointer P, processing in this step S209 is performed in order to make into radii the form of the locus

displayed with many locus images Lo. When a limiter is not applied to the value of the above-mentioned address pointer P by this processing, the form of the locus displayed with many locus images Lo will become a circle.

[0107] At step S210, 1d of operation means adds the number k of base addresses to an address pointer P. At step S211, the value of an address pointer P is assigned to the back address AD.

[0108] At step S212, 1m of guide information management means assigns a value only with less "k" than the value of an address pointer P to the front address ad. Here, the relation between the value which the back address AD shows, and the value which the front address ad shows is as being shown below.

AD<ad [0109] At step S213, it memorizes in the area where the value of the back address AD on main memory 5 shows the address data after conversion (x y) with which 1m of guide information management means is memorized in the area which the value of the front address ad on main memory 5 shows. At step S214, 1d of operation means subtracts the number k of base addresses from the front address ad.

[0110] At step S215, 1d of operation means subtracts the number k of base addresses from the back address AD. At step S216, it judges whether it is below the value that the addition result of having added the number k of base addresses to start-address Pstart shows, if the value which the front address ad shows is "YES", it will shift to step S217, and if 1f of decision means is "NO", they will shift to step S213 again.

[0111] In a main memory 5 top, the processing to the above-mentioned step S209 - step S216 is processing which moves the address data after conversion corresponding to the tail of an arrow head (x y) to the area corresponding to the address of a big value one by one, and, so to speak, is processing as a shift register. An address pointer P is substituted for the back address AD in step S211, and only k substitutes little data for the front address ad in step S213 for memorizing in the area where the value of the back address AD shows the address data after conversion (x y) memorized in the area which the value of the front address ad shows in step S212 from an address pointer P. That is, so to speak, this processing is processing as a shift register. Moreover, in step S214, the number k of base addresses is subtracted from the back address AD. In step S215, the number k of base addresses is subtracted from the front address ad. That the value which the front address ad shows in step S216 has judged whether it is below the value that the addition result of having added the number k of base addresses to start-address Pstart shows It is for judging whether there is any area which memorizes the address data after conversion (x y) memorized to the front address ad. That is, it is because there will be no storage area corresponding to the back address AD that the value of the front address ad turns into below the value that the above-mentioned addition result shows.

[0112] At step S217, 1m of guide information management means memorizes in the area where the back address AD of main memory 5 shows the address data after conversion of the top-most vertices of the flat surface corresponding to the tail of an arrow head (x y).

[0113] At step S218, 1h of variable setting means substitutes the addition result of start-address Pstart and the number k of base addresses for the 1st address AD 1. At step S219, 1h of variable setting means substitutes the addition result of start-address Pstart and double 2k of the number of base addresses for the 2nd address AD 2.

[0114] At step S220, 1m of guide information management means reads the address data after conversion on the main memory 5 which the 1st and 2nd address AD1 and AD2 shows (x y), and they supply the address data after the conversion concerned (x y) to the drawing processing processor 10 as a drawing instruction with texture address data and color palette data. The data memorized here in the area on the main memory 5 which the 1st address AD 1 shows are the after [conversion] address data equivalent to the tail of the arrow head currently displayed on this time. Moreover, the data memorized in the area on the main

memory 5 which the 2nd address AD 2 shows are the after [conversion] address data equivalent to the tail of the arrow head with which only 1 unit period was before displayed rather than this time. That is, the address data after conversion equivalent to the tail of the arrow head with which only 1 unit period was before displayed from this time and this time (x y) are supplied to the drawing processing processor 10 as address data of one image. Therefore, the drawing processing processor 10 writes in the texture data of an arrow head as texture data of the locus image Lo on the display area of a buffer 11 based on the address data after [of above-mentioned four] conversion (x y). The color of the locus image Lo at this time is the same as the color of the arrow head Na currently displayed at present. At step S221, 1d of operation means adds the number k of base addresses to the 1st address AD 1.

[0115] At step S222, operation means 1D adds the number k of base addresses to the 2nd address AD 2. At step S223, it judges whether it is beyond the value of an address pointer P, if the value of the 1st address AD 1 is "YES", it will escape from this throw guide display routine S200, and if 1f of decision means is "NO", they will shift to step S220 again.

[0116] The processing to the above-mentioned step S217 - step S223 is processing for supplying the address data after conversion (x y) memorized by main memory 5 to the drawing processing processor 10 as address data of 2 every area and the top-most vertices of one quadrilateral. In step S218, the addition result of start-address Pstart and the number k of base addresses is substituted for the 1st address AD 1. Having substituted the addition result of start-address Pstart and double 2k of the number of base addresses for the 2nd address AD 2 in step S219 It is for supplying the address data after conversion equivalent to the tail of one arrow head (x y), and the address data after conversion equivalent to the tail of the arrow head displayed before one rather than the above-mentioned arrow head (x y) to the drawing processing processor 10 as after [conversion] address data of one image.

[0117] Moreover, in step S221, the number k of base addresses is added to the 1st address AD 1. In step S222, the number k of base addresses is added to the 2nd address AD 2. That the value which the 1st address AD 1 shows in step S223 has judged whether it is more than the address pointer P It is for judging whether there are any address data after conversion (x y) corresponding to the 2nd address AD 2 used as the address data after conversion (x y) memorized to the 1st address AD 1 and a pair. That is, it is because there will be no address data after conversion corresponding to the 2nd address AD 2 that the value which the 1st address AD 1 shows becomes beyond the value which an address pointer P shows.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1] It is the game structure-of-a-system Fig. showing the gestalt of 1 operation of this invention.
[Drawing 2] It is the functional block diagram showing the function which CPU1 shown in drawing 1 has.
[Drawing 3] It is an explanatory view for explaining an arrow head and a locus.
[drawing 3 A] It is the explanatory view showing the relation between an orbit and an arrow head, and an example of a locus.
[drawing 3 B] It is the explanatory view showing an example of the absolute coordinate of an arrow head.
[drawing 3 C] It is an explanatory view for explaining the concept at the time of carrying out a false three-dimensional display from the absolute coordinate of an arrow head.
[Drawing 4] It is the explanatory view showing the condition of the image information on a table and main memory.
[drawing 4 A] It is the explanatory view showing the table which consists of arrow-head include-angle data corresponding to the number of coma, and this.
[drawing 4 B] It is the explanatory view showing the minimum value of data whenever [throw field and throw azimuth].
[drawing 4 C] It is the explanatory view showing an example of an arrow-head absolute coordinate.
[drawing 4 D] It is the explanatory view showing an example of the arrow-head address after conversion changed from the absolute coordinate shown in drawing 3 C.
[drawing 4 E] It is the explanatory view showing an example of a polygon absolute coordinate.

[drawing 4 F] It is the explanatory view showing an example of the polygon address after conversion changed from the absolute coordinate shown in drawing 3 E.
[drawing 4 G] It is the explanatory view showing an example of the posterior part address of Uchi of the arrow-head address after conversion shown in drawing 3 D.
[drawing 4 H] It is the explanatory view showing the address for locus generation obtained from the arrow-head posterior part address after [of four] conversion shown in drawing 3 G.
[Drawing 5] It is the explanatory view showing the example of a screen display of a hammer throw game as a gestalt of 1 operation.
[Drawing 6] It is a flow chart for explaining the control action by the main routine of a game program.
[Drawing 7] It is a flow chart for explaining the control action by the main routine of a game program.
[Drawing 8] It is a flow chart for explaining the control action by the main routine of a game program.

[Drawing 9] It is a flow chart for explaining the control action by the main routine of a game program.
[Drawing 10] It is a flow chart for explaining the control action by the polygon image display routine S100.
[Drawing 11] It is a flow chart for explaining the control action by the throw guide display routine S200.
[Drawing 12] It is a flow chart for explaining the control action by the throw guide display routine S200.
[Drawing 13] It is a flow chart for explaining the control action by the throw guide display routine S200.
[Description of Notations]
1 CPU
1a Button-grabbing detection means
1b View location data setting means
1c Display rectangle information extract means
1d Operation means
1e Result information setting means
1f Decision means
1g Drawing instruction issue means
1h Variable setting means
1i The number acquisition means of coma
1j Migration and a rotation acquisition means
1k Polygon information management means
1m Guide information management means
2 Bus
3 Graphics Data Generation Processor
4 20 Interface circuitry
5 Main Memory
6 ROM
7 Elongation Circuit
8 Parallel Port
9 Serial Port
10 Drawing Processing Processor
11, 14, 18 Buffer
13 Speech Processing Processor
15 Amplifying Circuit
16 Loudspeaker
17 Decoder
19 Record-Medium Driver
21 Memory
22 Controller

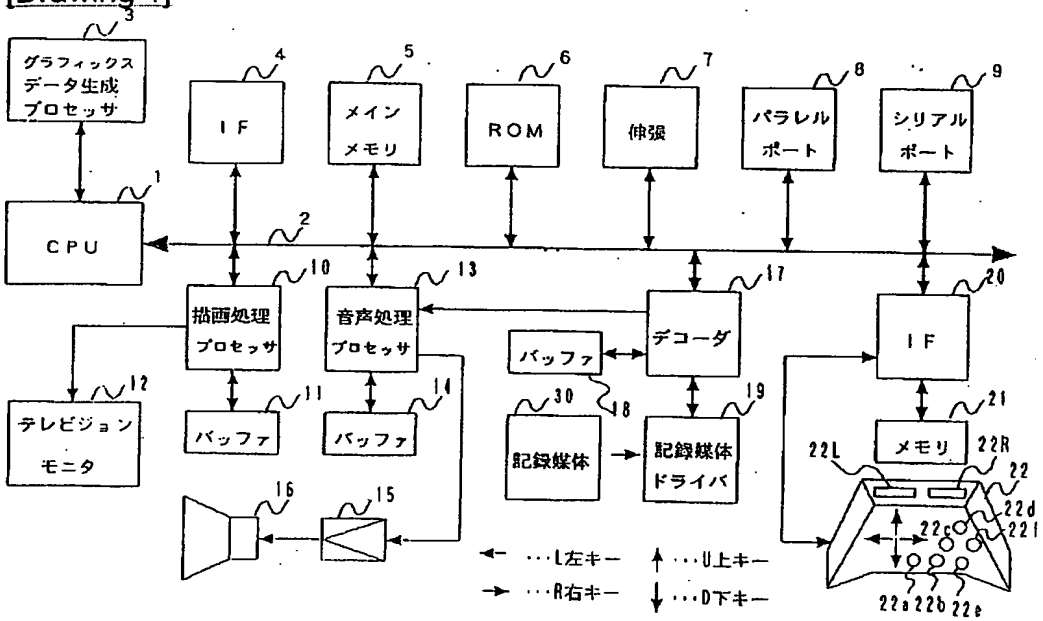
[Translation done.]

* NOTICES *

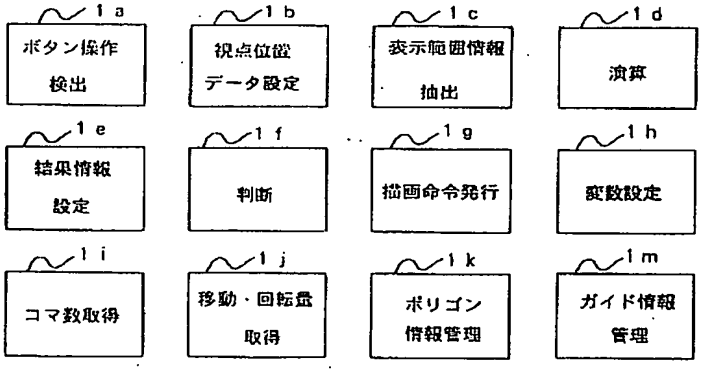
1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DRAWINGS

[Drawing 1]

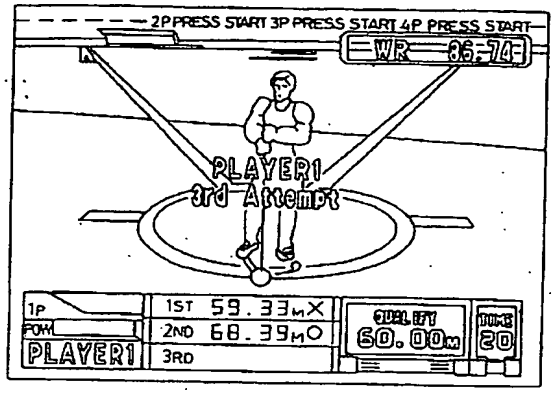


[Drawing 2]

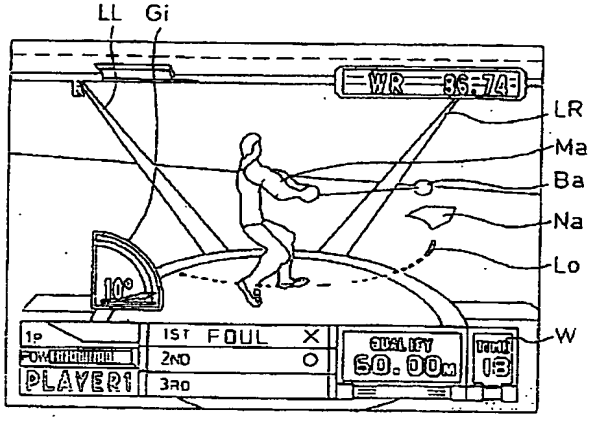


[Drawing 5]

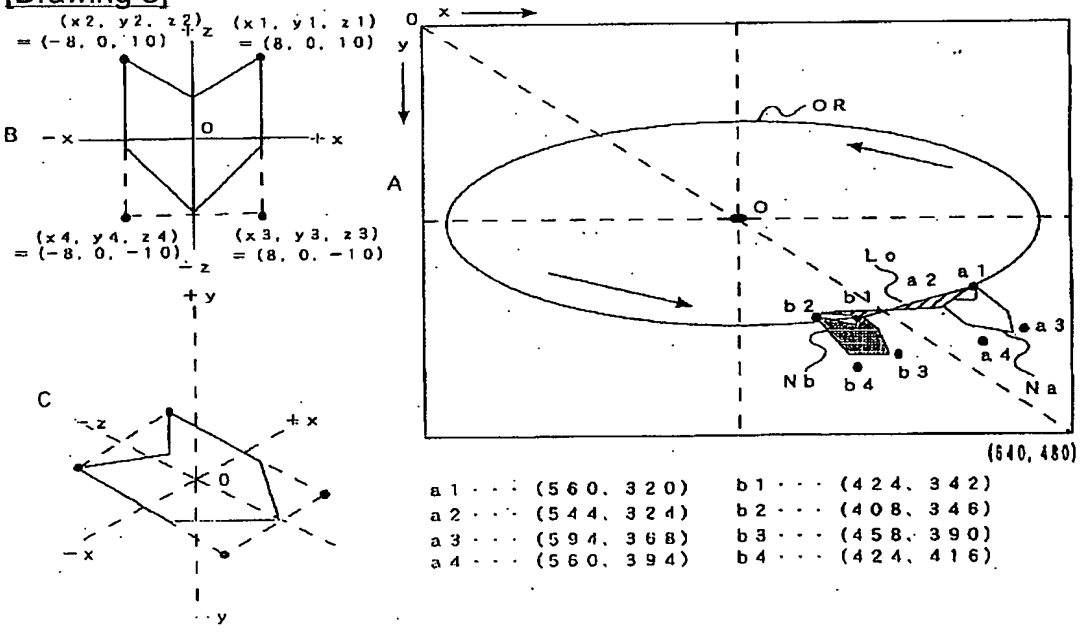
A



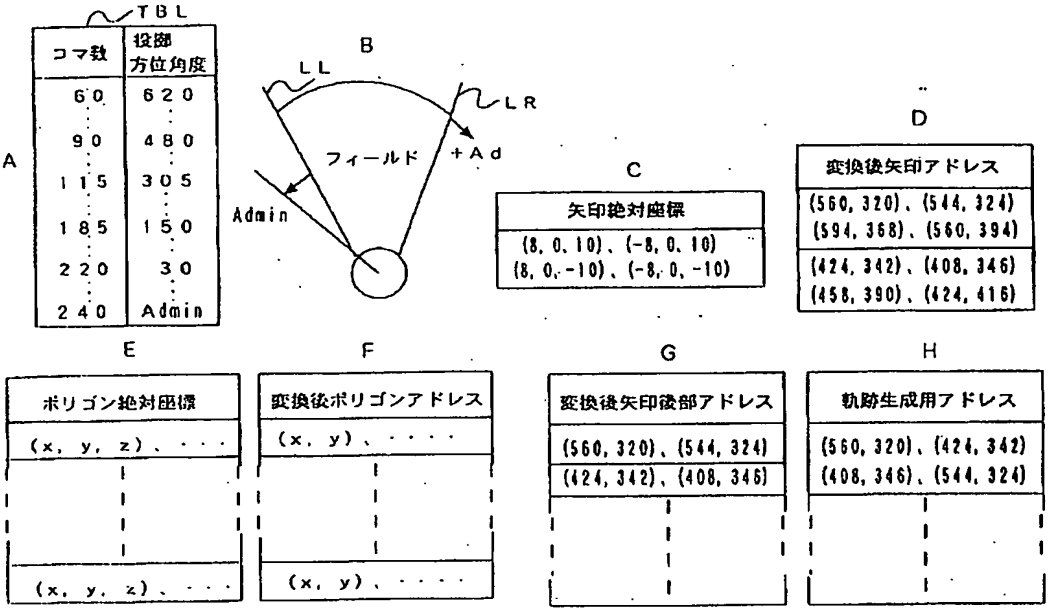
B



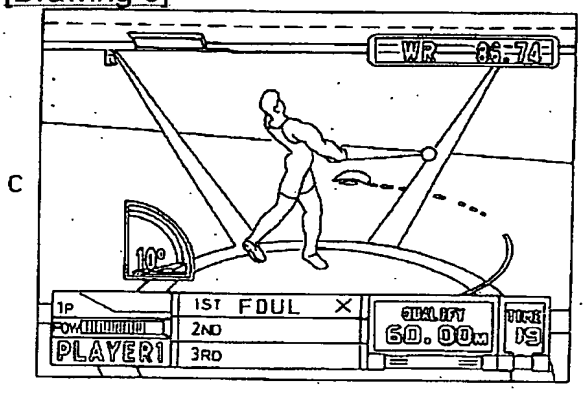
[Drawing 3]

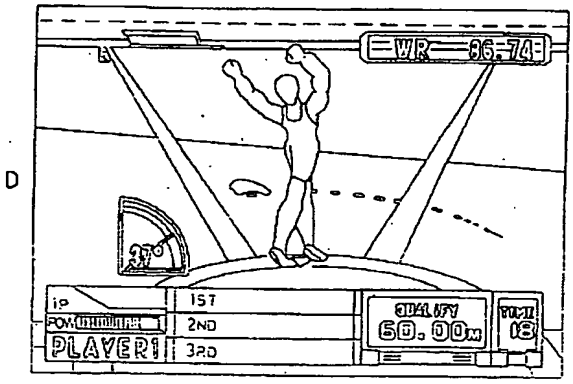


[Drawing 4]

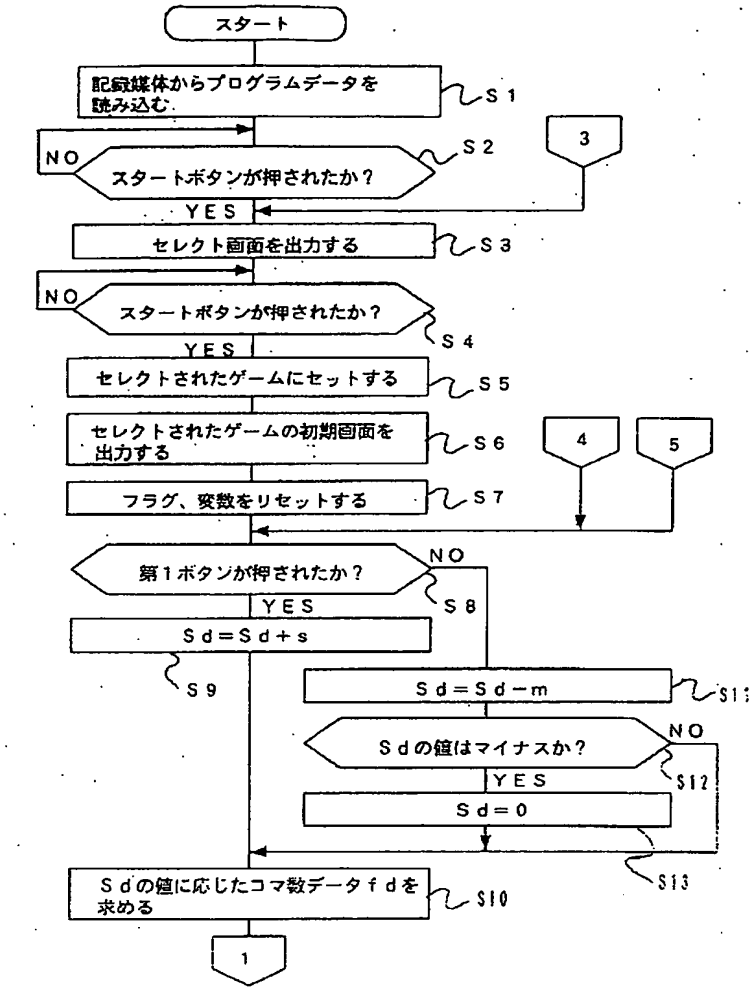


[Drawing 6]

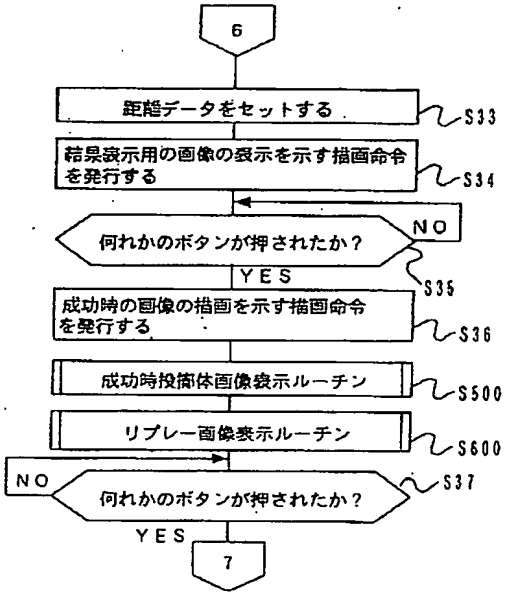




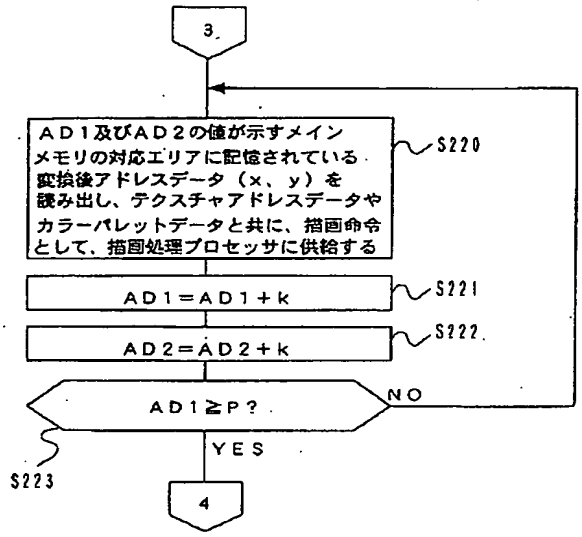
[Drawing 7]


スタート
記録媒体からプログラムデータを読み込む
S1
NO
スタートボタンが押されたか？
S2
3
YES
セレクト画面を出力する
S3
NO
スタートボタンが押されたか？
S4
YES
セレクトされたゲームにセットする
S5
セレクトされたゲームの初期画面を出力する
S6
4
5
フラグ、変数をリセットする
S7
第1ボタンが押されたか？
NO
S8
YES
Sd＝Sd＋s
S9
Sd＝Sd－m
S11
Sdの値はマイナスか？
NO
S12
YES
Sd＝0
S13
Sdの値に応じたコマ数データfdを求める
S10
1

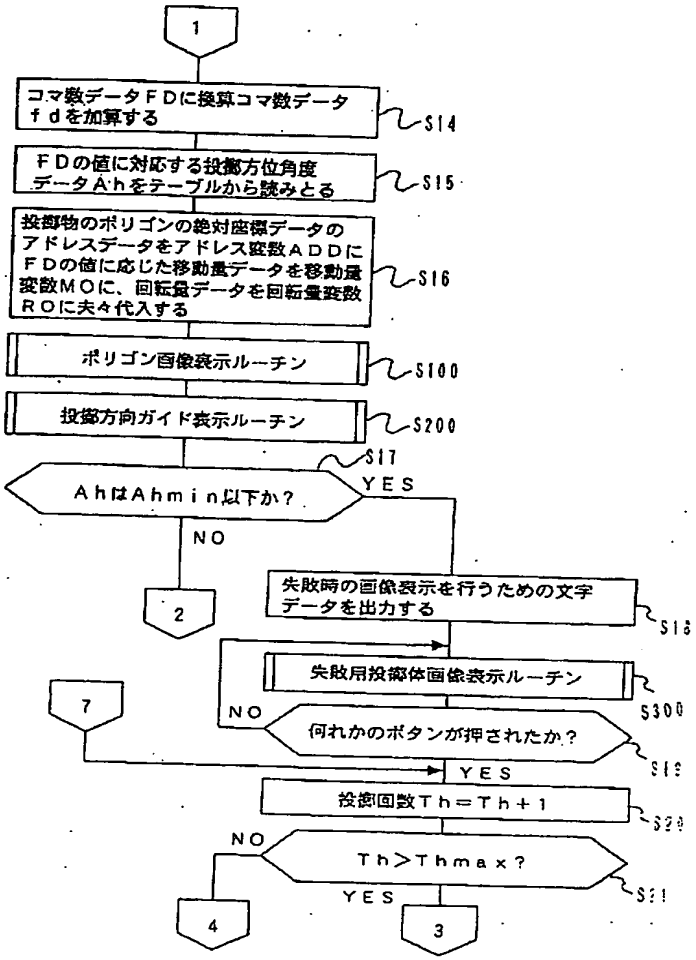
[Drawing 10]


6
距離データをセットする
S33
背景表示用の画像の表示を示す描画命令を発行する
S34
何れかのボタンが押されたか？
NO
S35
YES
成功時の画像の描画を示す描画命令を発行する
S36
成功時投擲体画像表示ルーチン
S500
リプレー画像表示ルーチン
S600
NO
何れかのボタンが押されたか？
S37
YES
7

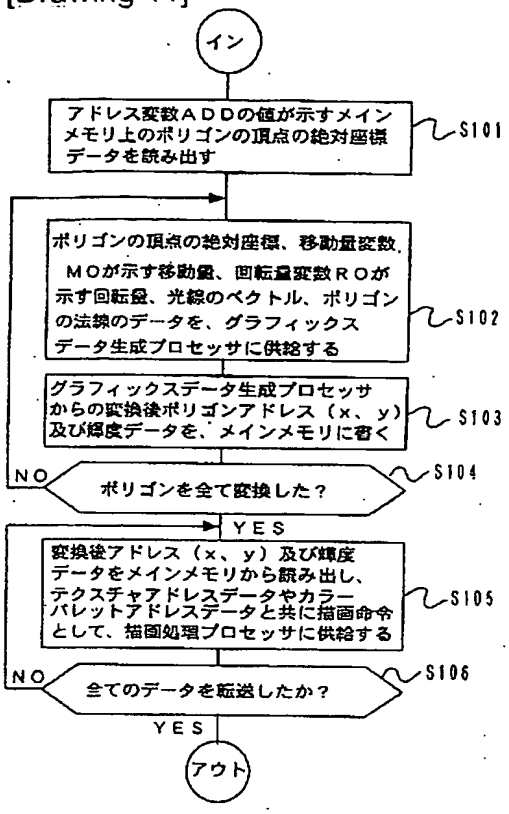
[Drawing 14]



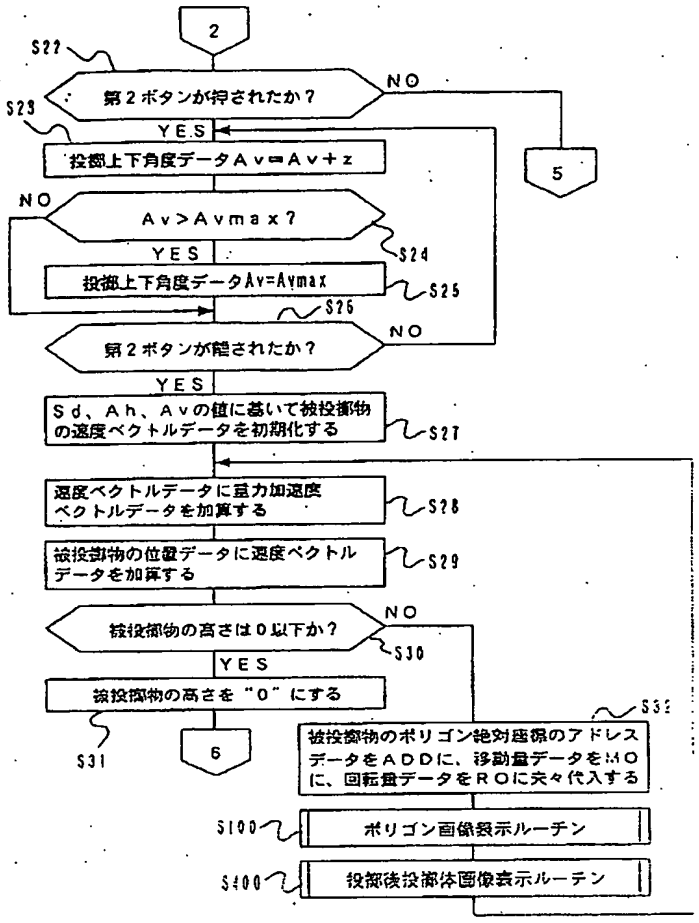
[Drawing 8]



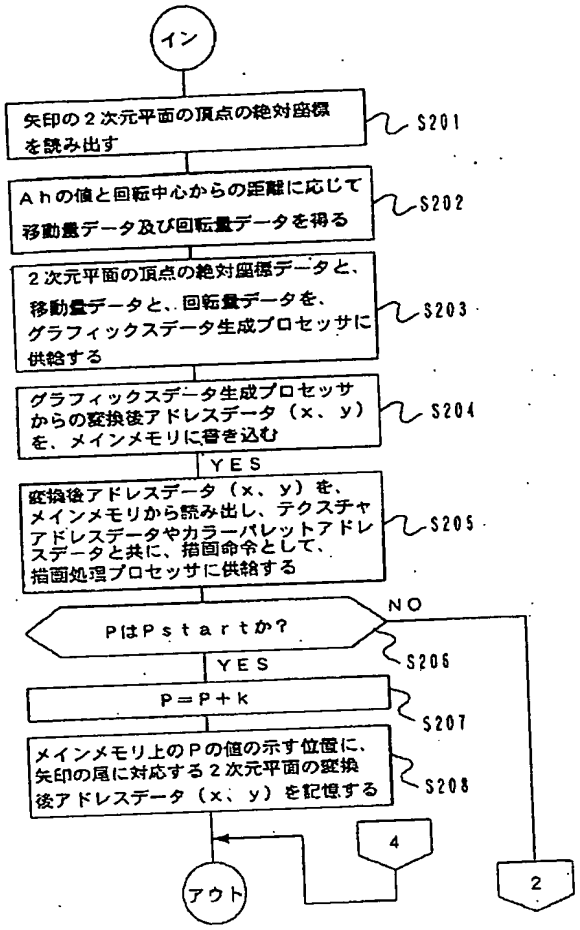
[Drawing 11]



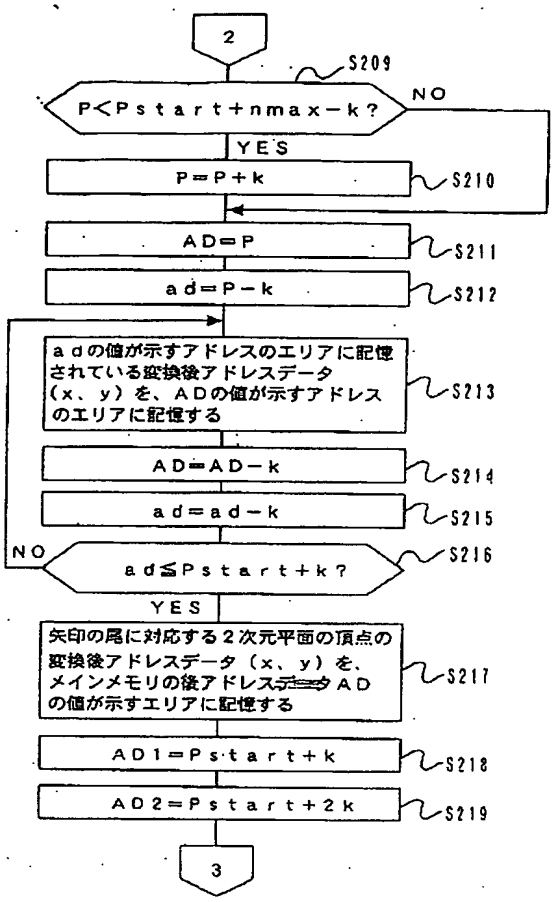
[Drawing 9]


2
S22
第２ボタンが押されたか？
NO
S23
YES
投擲上下角度データＡｖ＝Ａｖ＋ｚ
5
NO
Ａｖ＞Ａｖｍａｘ？
YES
S24
投擲上下角度データAv=Avmax
S25
S26
第２ボタンが離されたか？
NO
YES
Ｓｄ、Ａｈ、Ａｖの値に基いて被投擲物の速度ベクトルデータを初期化する
S27
速度ベクトルデータに重力加速度ベクトルデータを加算する
S28
被投擲物の位置データに速度ベクトルデータを加算する
S29
被投擲物の高さは０以下か？
NO
S30
YES
被投擲物の高さを“０”にする
S31
6
S32
被投擲物のポリゴン絶対座標のアドレスデータをＡＤＤに、移動量データをＭＯに、回転量データをＲＯに次々代入する
S100
ポリゴン画像表示ルーチン
S400
投擲後投擲体画像表示ルーチン

[Drawing 12]


イン
矢印の２次元平面の頂点の絶対座標を読み出す
S201
Ａｈの値と回転中心からの距離に応じて移動量データ及び回転量データを得る
S202
２次元平面の頂点の絶対座標データと、移動量データと、回転量データを、グラフィックスデータ生成プロセッサに供給する
S203
グラフィックスデータ生成プロセッサからの変換後アドレスデータ（ｘ、ｙ）を、メインメモリに書き込む
S204
ＹＥＳ
変換後アドレスデータ（ｘ、ｙ）を、メインメモリから読み出し、テクスチャアドレスデータやカラーパレットアドレスデータと共に、描画命令として、描画処理プロセッサに供給する
S205
ＰはＰｓｔａｒｔか？
NO
S206
ＹＥＳ
Ｐ＝Ｐ＋ｋ
S207
メインメモリ上のＰの値の示す位置に、矢印の属に対応する２次元平面の変換後アドレスデータ（ｘ、ｙ）を記憶する
S208
4
2
アウト

[Drawing 13]



[Translation done.]

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（A）**　(11)特許出願公開番号

特開平10－52572

(43)公開日　平成10年(1998) 2月24日

(51)Int.Cl.[6]	識別記号	庁内整理番号	ＦＩ		技術表示箇所
Ａ６３Ｆ　9/22			Ａ６３Ｆ　9/22	Ｐ	
				Ｈ	

審査請求　未請求　請求項の数38　ＦＤ　（全 24 頁）

(21)出願番号　特願平9－147052

(22)出願日　平成９年(1997) ５月22日

(31)優先権主張番号　特願平8－149714

(32)優先日　平８(1996) ５月22日

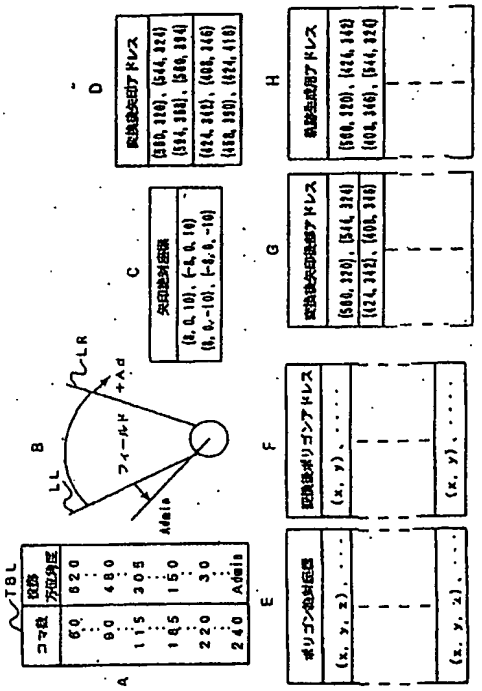
(33)優先権主張国　日本（ＪＰ）

(71)出願人　000105637
コナミ株式会社
兵庫県神戸市中央区港島中町７丁目３番地の２

(72)発明者　大島　直人
兵庫県神戸市中央区港島中町７丁目３番地の２　コナミ株式会社内

(54)【発明の名称】　投擲ゲームにおける投擲ガイド表示方法、ゲームシステム及び記録媒体

(57)【要約】
【課題】　被投擲物の投擲タイミングや投擲方向を、ゲームプレーヤに認識し易くすることを課題とする。
【解決手段】　ハンマー投げ、砲丸投げ、円盤投げなどのように、選手が回転して被投擲物を投擲するゲームにおいて、投擲方向と回転数を示す投擲ガイドを表示面上に表示する。



1

【特許請求の範囲】

【請求項１】　少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像とが表示手段の表示面上に表示され、操作手段の操作に基いて、上記投擲体が視覚的に上記被投擲物を投擲するために動作させられると共に、上記操作手段の操作に基いて、上記被投擲物が上記表示手段の表示面上において上記投擲体により視覚的に投擲される投擲ゲームで用いられる、投擲ゲームにおける投擲ガイド表示方法であって、

上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示する投擲ゲームにおける投擲ガイド表示方法。

【請求項２】　上記投擲体が上記被投擲物を投擲するための動作は、弧を描く動作若しくは略回転動作である請求項１記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項３】　上記投擲ガイド画像は、上記投擲体の動作に応じた軌道を描くように順次表示される請求項２記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項４】　上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示される請求項２記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項５】　上記軌跡画像は、現在の投擲ガイド画像の座標情報と、所定時間前に表示された投擲ガイド画像の座標情報に基いて生成される請求項４記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項６】　上記投擲ガイド画像は、上記表示手段の表示面上において、疑似３次元表示される請求項１記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項７】　上記投擲ガイド画像の形状は、少なくとも矢印形状若しくはこれに類する、視覚的方向指示機能を有する形状である請求項１記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項８】　少なくとも上記投擲体の回転に応じて、上記投擲ガイド画像の色が可変される請求項１記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項９】　上記投擲ガイド画像の色は、上記投擲体の回転の進行に従って、寒色から暖色に変化される請求項８記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１０】　少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像とが表示手段の表示面上に表示され、操作手段の操作に基いて、上記投擲体が視覚的に上記被投擲物を投擲するために動作させられると共に、上記操作手段の操作に基いて、上記被投擲物が上記表示手段の表示面上において上記投擲体により視覚的に投擲される投擲ゲームで用いられる、投擲ゲームにおける投擲ガイド表示方法であって、

2

上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示する投擲方向表示ステップを含む投擲ゲームにおける投擲ガイド表示方法。

【請求項１１】　上記投擲方向表示ステップは、

上記操作手段の操作に基いた上記被投擲物の投擲角度情報に基づく上記投擲ガイド画像の移動量及び回転量を求める移動・回転量取得ステップと、

上記投擲ガイドの座標情報と、上記移動・回転量取得ステップにおいて求められた移動量及び回転量情報とに基いて、上記投擲ガイドのメモリ上におけるアドレスデータを取得するアドレスデータ取得ステップと、

少なくとも、上記投擲ガイド画像の代表点のメモリ上のアドレスデータと、上記投擲ガイドの色を指定するための情報と、上記投擲ガイドのテクスチャ情報とを、描画処理手段に供給することにより描画を指示する描画指示ステップとを含む請求項１０記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１２】　少なくとも上記投擲体の回転に応じて、上記投擲ガイド画像の色が可変される請求項１１記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１３】　上記投擲ガイド画像の色は、上記投擲体の回転の進行に従って、寒色から暖色に変化するよう順次指定される請求項１２記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１４】　上記投擲ガイドの軌跡を示す軌跡画像を生成する軌跡生成ステップを更に設け、

当該軌跡生成ステップは、

投擲ガイド画像のメモリ上のアドレスデータを記憶する記憶ステップと、

少なくとも、現在及び過去の２つの投擲ガイド画像のメモリ上のアドレスデータから抽出した１つ分のアドレスデータと、色を指定するための情報と、上記ガイド画像のテクスチャ情報とを、描画処理手段に供給することにより、上記軌跡画像の描画を指示する軌跡画像描画指示ステップとを含む請求項１０記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１５】　上記投擲体が上記被投擲物を投擲するための動作は、弧を描く動作若しくは略回転動作である請求項１０記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１６】　上記投擲ガイド画像は、上記投擲体の動作に応じた軌道を描くように順次表示される請求項１５記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１７】　上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示される請求項１５記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１８】　上記投擲ガイド画像は、上記表示手段の表示面上において、疑似３次元表示される請求項１０

記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項１９】　上記投擲ガイド画像の形状は、少なくとも矢印形状若しくはこれに類する視覚的方向指示機能を有する形状である請求項１０記載の投擲ゲームにおける投擲ガイド表示方法。

【請求項２０】　少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像情報とを表示する表示手段と、操作手段と、上記操作手段の操作に基いて、上記表示手段の表示面上において上記投擲体が上記被投擲物を投擲するための動作、並びに上記被投擲物の投擲を、視覚的に行う制御手段とを備える、ゲームシステムであって、

上記制御手段は、

上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示する投擲方向表示手段を有するゲームシステム。

【請求項２１】　上記投擲体が上記被投擲物を投擲するための動作は、弧を描く動作若しくは略回転動作である請求項２０記載のゲームシステム。

【請求項２２】　上記投擲ガイド画像は、上記投擲体の動作に応じた軌道を描くように順次表示される請求項２１記載のゲームシステム。

【請求項２３】　上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示される請求項２１記載のゲームシステム。

【請求項２４】　上記軌跡画像は、現在の投擲ガイド画像の座標情報と、所定時間前に表示された投擲ガイド画像の座標情報に基いて生成される請求項２３記載のゲームシステム。

【請求項２５】　上記投擲ガイド画像は、上記表示手段の表示面上において、疑似３次元表示される請求項２０記載のゲームシステム。

【請求項２６】　上記投擲ガイド画像の形状は、少なくとも矢印形状若しくはこれに類する、視覚的方向指示機能を有する形状である請求項２０記載のゲームシステム。

【請求項２７】　少なくとも上記投擲体の回転に応じて、上記投擲ガイド画像の色が可変される請求項２０記載のゲームシステム。

【請求項２８】　上記投擲ガイド画像の色は、上記投擲体の回転の進行に従って、寒色から暖色に変化させられる請求項２７記載のゲームシステム。

【請求項２９】　少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像とが表示手段の表示面上に表示され、操作手段の操作に基いて、上記投擲体が視覚的に上記被投擲物を投擲するために動作させられると共に、上記操作手段の操作に基いて、上記被投擲物が上記表示手段の表示面上において上記投擲体により視覚的に投擲されるゲームプログラムを含む投擲ゲームデータが記録された記録媒体であって、

上記ゲームプログラムは、

上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示する投擲方向表示ステップを含む記録媒体。

【請求項３０】　上記投擲方向表示ステップは、

上記操作手段の操作に基いた上記被投擲物の投擲角度情報に基づく上記投擲ガイド画像の移動量及び回転量を求める移動・回転量取得ステップと、

上記投擲ガイドの座標情報と、上記移動・回転量取得ステップにおいて求められた移動量及び回転量情報とに基いて、上記投擲ガイドのメモリ上におけるアドレスデータを取得するアドレスデータ取得ステップと、

少なくとも、上記投擲ガイド画像の代表点のメモリ上のアドレスデータと、上記投擲ガイドの色を指定するための情報と、上記投擲ガイドのテクスチャ情報とを、描画処理手段に供給することにより描画を指示する描画指示ステップとを含む請求項２９記載の記録媒体。

【請求項３１】　少なくとも上記投擲体の回転に応じて、上記投擲ガイド画像の色が可変される請求項２９記載の記録媒体。

【請求項３２】　上記投擲ガイド画像の色は、上記投擲体の回転の進行に従って、寒色から暖色に変化するよう順次指定される請求項３１記載の記録媒体。

【請求項３３】　上記投擲ガイドの軌跡を示す軌跡画像を生成する軌跡生成ステップを更に設け、

当該軌跡生成ステップは、

投擲ガイド画像のメモリ上のアドレスデータを記憶する記憶ステップと、

少なくとも、現在及び過去の２つの投擲ガイド画像のメモリ上のアドレスデータから抽出した１つ分のアドレスデータと、色を指定するための情報と、上記ガイド画像のテクスチャ情報とを、描画処理手段に供給することにより上記軌跡画像の描画を指示する軌跡画像描画指示ステップとを含む請求項２９記載の記録媒体。

【請求項３４】　上記投擲体が上記被投擲物を投擲するための動作は、弧を描く動作若しくは略回転動作である請求項２９記載の記録媒体。

【請求項３５】　上記投擲ガイド画像は、上記投擲体の動作に応じた軌道を描くように順次表示される請求項３４記載の記録媒体。

【請求項３６】　上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示される請求項３４記載の記録媒体。

【請求項３７】　上記投擲ガイド画像は、上記表示手段の表示面上において、疑似３次元表示される請求項２９記載の記録媒体。

【請求項３８】　上記投擲ガイド画像の形状は、少なく

とも矢印形状若しくはこれに類する視覚的方向指示機能を有する形状である請求項２９記載の記録媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、ゲームデータの記録された、光ディスク、磁気ディスク、半導体メモリを用いたカセット式記録媒体等を用いるゲームシステムに適用して好適な投擲ゲームにおける投擲ガイド表示方法、ゲームシステム及び記録媒体に関する。

【０００２】

【従来の技術】ゲームシステムは数多く提案されている。家庭用の専用機とテレビジョンモニタとからなるシステム、業務用の専用機、パーソナルコンピュータ若しくはワークステーションとディスプレイと音声出力機とからなるシステム等である。これらのシステムは、何れも、プレーヤが操作するためのコントローラと、ゲームプログラムデータと画像や音声などのデータからなるゲームデータの記録された記録媒体と、ゲームプログラムデータに基いて音声や画像の生成のための制御を行うＣＰＵと、画像を処理するためのプロセッサと、音声を処理するためのプロセッサと、画像を表示するためのＣＲＴと、音声を出力するためのスピーカとで構成される。上記記録媒体としては、ＣＤ－ＲＯＭ、半導体メモリ、半導体メモリを内蔵したカセット等が多い。ゲームシステムの構成は以上の通りである。

【０００３】一方、ゲームの種類は増加の一途をたどり、また、ゲームの内容は、日増しに複雑、且つ、多様化してきている。最近では、コントローラを操作して画面上の選手を動かすことにより、テレビジョンモニタの表示面上に形成されるゲーム空間上で、擬似的にスポーツを行うものまで提案されている。スポーツゲームとして、サッカーや野球等の団体競技ゲームや、個人競技ゲームが提案されている。個人競技ゲームとしては、大別すると、「走る競技」、「泳ぐ競技」、「選手自身が飛ぶ競技」、「物を持ち上げる競技」、「格闘する競技」、「的に物を当てる競技」、「選手自身が物を投げる（投擲）競技」がある。尚、投擲競技としては、「砲丸投げ」、「ハンマー投げ」、「円盤投げ」、「やり投げ」等がある。これらの個人競技を、テレビゲームとして実現した場合、その形態は、次のようになることは予測できよう。即ち、ゲームシステムの制御部が、ゲームプレーヤがコントローラを操作したときに、その操作内容、操作状態に基いて、テレビジョンモニタの表示面上に形成されたゲーム空間内の選手等を視覚的に動かして、競技を行わせる。

【０００４】

【発明が解決しようとする課題】上記投擲競技の内、砲丸投げは、選手が略１８０度回転した後に球を投げることにより行われる。円盤投げやハンマー投げは、選手が競技エリア内において複数回回転した後に円盤やハンマーを投げることにより行われる。このような投擲競技をゲームとして実現するためには、少なくとも、コントローラの何らかの操作により選手の投擲エネルギーを設定する処理と、表示面上において、選手の画像を、半回転（砲丸投げ）若しくは複数回転しているように表示する処理と、半回転若しくは複数回転しているように表示しているときに、コントローラの何らかの操作により、ゲーム空間内の選手が被投擲物を投擲するように表示する処理と、上記投擲エネルギー、投擲方向及び角度に基いて、ゲーム空間内において被投擲物が飛ぶように表示する処理が必要である。ゲーム空間内で行われる競技を、視覚的に、より本物の競技により近づけるためである。

【０００５】しかしながら、被投擲物の飛ぶ方向を、ゲーム空間内の選手の姿勢で決めるようにすると、ゲームプレーヤは、ゲーム空間内の選手が被投擲物を投擲するタイミングを判断することが困難となる。画面上に順次表示される選手画像の状態だけで、ゲーム空間内の選手が被投擲物を投擲するタイミングを判断しなければならないからである。

【０００６】また、ハンマー投げの場合においては、ゲーム空間内の選手の腕の画像によって示される予測ベクトルと、この選手によって振り回されているハンマーの画像によって示される予測ベクトルは、異なる。例えばモーション・キャプチャの手法を用いて、被投擲物を投擲する選手の動画像を作成した場合、慣性は、視覚的であっても、本物と同様に表現される。つまり、ハンマーの画像は、回転している選手の腕の画像よりも回転方向において手前となる。従って、ゲームプレーヤは、益々、ゲーム空間内の選手にハンマーを離させるタイミングをつかみにくくなる。

【０００７】更に、ハンマー投げや円盤投げ等のように、投擲体を複数回回転させて被投擲物を投擲するような競技は、主流となっている回転数がある。よって、ゲームにおいても、投擲体の回転数が、主流となっている回転数を越えたときに、未だ被投擲物が投擲されていない場合には、失敗等にする必要がある。ゲーム性を確保するためでもある。しかしながら、あと何回転投擲体を回転させて良いのかを投擲体が回転する状態を見ながらカウントすることは、ゲームプレーヤの数え間違いを招くと共に、ゲームプレーヤのゲームに対する集中力の低下を招き、ひいては、ゲームの結果に影響を及ぼす虞がある。

【０００８】本発明はこのような点を考慮してなされたもので、ゲームプレーヤに対し、被投擲物の投擲方向、残りの回転数を、分かり易くガイドすることを目的とする。

【０００９】

【課題を解決するための手段】本発明の主要な１つは、少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像とが表示手段

の表示面上に表示され、操作手段の操作に基いて、上記投擲体が視覚的に上記被投擲物を投擲するために動作させられると共に、上記操作手段の操作に基いて、上記被投擲物が上記表示手段の表示面上において上記投擲体により視覚的に投擲される、投擲ゲームにおける投擲ガイド表示方法であって、上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示するものである。

【００１０】また本発明の他の主要な１つは、上記発明において、上記投擲体が上記被投擲物を投擲するための動作が、弧を描く動作若しくは略回転動作であるものである。

【００１１】また本発明の他の主要な１つは、上記発明において、上記投擲ガイド画像が、上記投擲体の動作に応じた軌道を描くように順次表示されるものである。

【００１２】また本発明の他の主要な１つは、上記発明において、上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示されるものである。

【００１３】また本発明の他の主要な１つは、上記発明において、上記軌跡画像が、１つの投擲ガイド画像の座標情報と、当該投擲ガイド画像よりも１つ前に表示された投擲ガイド画像の座標情報に基いて生成されるものである。

【００１４】また本発明の他の主要な１つは、上記発明において、上記投擲ガイド画像が、上記表示手段の表示面上において、疑似３次元表示されるものである。

【００１５】また本発明の他の主要な１つは、上記発明において、上記投擲ガイド画像の形状が、少なくとも矢印形状若しくはこれに類する、視覚的方向指示機能を有する形状であるものである。

【００１６】また本発明の他の主要な１つは、上記発明において、少なくとも、上記投擲体の回転数に応じて、上記投擲ガイド画像の色が可変されるものである。

【００１７】また本発明の他の主要な１つは、上記発明において、上記投擲ガイド画像の色が、上記投擲体の上記投擲体の回転数の値が小さい値から大きい値に変換されるのに従って、寒色から暖色に変化されるものである。

【００１８】

【発明の実施の形態】以下に、図１～図１３を順次参照して本発明の実施の形態について詳細に説明する。

【００１９】本発明の実施の形態の説明は、次に示す項目説明を各項目の先頭に記載し、各項目について次に示す順序で説明する。

【００２０】Ａ．ゲームシステムの構成（図１）
Ｂ．図１に示したＣＰＵ１の有する機能（図２）
Ｃ．矢印及び軌跡の表示（図３並びに図４Ｃ、Ｄ、Ｇ、及びＨ）
Ｄ．画像表示例（図５及び図６）
Ｅ．メインルーチンによる制御動作（図７～図１０）
Ｆ．ポリゴン画像表示ルーチンＳ１００による制御動作（図１１）
Ｇ．投擲ガイド表示ルーチンＳ２００による制御動作（図１２～図１４）

【００２１】Ａ．ゲームシステムの構成（図１）

【００２２】図１は、本発明の一実施の形態としてのゲームシステムの構成例を示す構成図である。

【００２３】〔接続及び構成〕この図１に示されるゲームシステムは、ゲーム機本体と、画像や音声、並びにプログラムデータからなるゲームデータの記録された記録媒体３０とからなる。ゲーム機本体は、ＣＰＵ１と、このＣＰＵ１に、アドレス、データ及びコントロールバスからなるバス２が接続され、このバス２に、グラフィックスデータ生成プロセッサ３、インターフェース回路４、メインメモリ５、ＲＯＭ６、伸張回路７、パラレルポート８、シリアルポート９、描画処理プロセッサ１０及びバッファ１１、音声処理プロセッサ１３及びバッファ１４、デコーダ１７及びバッファ１８、インターフェース回路２０及びメモリ２１とが夫々接続され、更に、描画処理プロセッサ１０にテレビジョンモニタ１２が接続され、音声処理プロセッサ１３に増幅回路１５を介してスピーカ１６が接続され、デコーダ１７に記録媒体ドライバ１９が接続され、インターフェース回路２０にコントローラ２２が接続されて構成される。

【００２４】ここで、上記ゲームシステムは、用途に応じてその形態が異なる。即ち、上記ゲームシステムが、家庭用として構成されている場合においては、テレビジョンモニタ１２及びスピーカ１６は、ゲーム機本体とは別体となる。また、上記ゲームシステムが、業務用として構成されている場合においては、図１に示されている構成要素はすべて一体型となっている１つの筐体に収納される。また、上記ゲームシステムが、パーソナルコンピュータやワークステーションを核として構成されている場合においては、上記テレビジョンモニタ１２は、上記コンピュータ用のディスプレイに対応し、上記描画処理プロセッサ１０、音声処理プロセッサ１３、伸張回路７は、夫々上記記録媒体３０に記録されているゲームプログラムデータの一部若しくはコンピュータの拡張スロットに搭載される拡張ボード上のハードウエアに対応し、上記インターフェース回路４、上記パラレルポート８、上記シリアルポート９及びインターフェース回路２０は、コンピュータの拡張スロットに搭載される拡張ボード上のハードウエアに対応する。また、上記バッファ１１、１４及び１８は、夫々メインメモリ５若しくは図示しない拡張メモリの各エリアに対応する。本形態では、上記ゲームシステムが、家庭用として構成されている場合を例にとり説明する。

【００２５】次に、図１に示した各構成要素についてより詳細に説明する。グラフィックスデータ生成プロセッ

サ３は、ＣＰＵ１のいわばコプロセッサとしての役割を果たす。即ち、このグラフィックスデータ生成プロセッサ３は、座標変換や光源計算、例えば固定小数点形式の行列やベクトルの演算を、並列処理により行う。このグラフィックスデータ生成プロセッサ３の主な処理は、座標変換処理及び光源計算処理である。座標変換処理は、ＣＰＵ１から供給される画像データの２次元若しくは３次元面内における各頂点の絶対座標データを、移動量データ及び回転量データに基いて、処理対象画像の表示エリア上におけるアドレスを求め、当該アドレスデータを、再びＣＰＵ１に返す処理である。この座標変換処理については後に詳述する。

【００２６】また、光源計算処理は、光線のベクトルデータと、ポリゴンの面の向きを表す法線データと、面の色を示すデータとに応じて、画像の輝度を計算する処理である。

【００２７】上記インターフェース回路４は、周辺デバイス、例えばマウスやトラックボール等のポインティングデバイス等のインターフェース用である。上記ＲＯＭ６にはゲームシステムのオペレーティングシステムとしてのプログラムデータが、記憶されている。パーソナルコンピュータで言えば、ＢＩＯＳ（Ｂａｓｉｃ　Ｉｎｐｕｔ　Ｏｕｔｐｕｔ　Ｓｙｓｔｅｍ）に相当する。

【００２８】上記伸張回路７においては、ＭＰＥＧ（Ｍｏｖｉｎｇ　Ｐｉｃｔｕｒｅ　Ｅｎｇｉｎｅｅｒｉｎｇ　Ｇｒｏｕｐ）やＪＰＥＧ（Ｊｏｉｎｔ　ＰｉｃｔｕｒｅＥｎｇｉｎｅｅｒｉｎｇ　Ｇｒｏｕｐ）に準拠したイントラ符号化により圧縮された圧縮画像に対し、伸張処理が施される。伸張処理は、デコード処理（ＶＬＣ：Ｖａｒｉａｖｌｅ　Ｌｅｎｇｔｈ　Ｃｏｄｅによりエンコードされたデータのデコード）、逆量子化処理、ＩＤＣＴ（Ｉｎｖｅｒｓｅ　Ｄｉｓｃｒｅｔｅ　Ｃｏｓｉｎｅ　Ｔｒａｎｓｆｏｒｍ）処理、イントラ画像の復元処理等である。

【００２９】描画処理プロセッサ１０は、ＣＰＵ１が発行する描画命令に基いて、バッファ１１に対する描画処理を行う。バッファ１１は、表示エリアと非表示エリアとからなる。表示エリアは、テレビジョンモニタ１２の表示面上に表示されるデータの展開エリアである。非表示エリアは、テクスチャデータやカラーパレットデータ等の記憶エリアである。ここで、テクスチャデータは、２次元の画像データである。カラーパレットデータは、テクスチャデータ等の色を指定するためのデータである。これらのデータは、ＣＰＵ１により、記録媒体３０から１回、若しくはゲームの進行状況に応じて複数回に分けて読み出され、予めバッファ１１の非表示エリアに記憶される。

【００３０】描画命令としては、例えばラインを描画するための描画命令、ポリゴンを用いて立体的な画像を描画するための描画命令、通常の２次元画像を描画するための描画命令がある。ここで、ポリゴンは、多角形の２次元画像であり、本形態においては、三角形若しくは四角形が用いられる。

【００３１】ラインを描画するための描画命令は、ラインの描画開始及び終了アドレス、色及びライン描画を意味するデータからなる。このライン描画命令は、ＣＰＵ１により、直接、描画処理プロセッサ１０に対して発行される。

【００３２】ポリゴンを用いて立体的な画像を描画するための描画命令は、バッファ１１の表示エリア上におけるポリゴン頂点アドレスデータ、ポリゴンに貼り付けるテクスチャデータのバッファ１１上における記憶位置を示すテクスチャアドレスデータ、テクスチャデータの色を示すカラーパレットデータのバッファ１１上における記憶位置を示すカラーパレットアドレスデータ並びにテクスチャの輝度を示す輝度データとからなる。これらのデータの内、ポリゴン頂点アドレスデータは、グラフィックスデータ生成プロセッサ３が、ＣＰＵ１からの、ポリゴン絶対座標データと、ポリゴンの動きを示すデータと、視点位置の動きを示すデータとに基いて演算を行うことによって得られる。ここで、ポリゴン頂点アドレスデータがどのようにして求められるのかについて説明する。

【００３３】テレビジョンモニタ１２の表示面上における物体の動きは、物体のそのものの動きと、この物体に対する視点位置の動きとで決まる。例えば、物体のみが動き、視点位置が固定されているものとすれば、テレビジョンモニタ１２の表示面上における物体の動きは、物体そのものの動きである。逆に、物体に動きがなく、視点位置のみが動かされたものとすれば、テレビジョンモニタ１２の表示面上における動きは、視点位置そのものの動きである。尚、「視点位置」を、「カメラ位置」と読み替えると、より理解し易いであろう。つまり、テレビジョンモニタ１２の表示面上においては、あたかも、カメラを動かしながら物体を撮像したような表示が行われる。説明を簡単にするために、物体若しくは視点位置のどちらか一方が動く場合について説明したが、通常は、物体及び視点位置の両方が動いているように処理され、その結果が表示される。

【００３４】ここで、上記物体の「動き」は、「回転量」及び「移動量」からなる。視点位置に対する物体の回転量は、物体の回転角と、視点位置の回転角とで生成される。ここで、回転量、回転角は、２次元座標系が用いられる処理では２×２、３次元座標系が用いられる処理では３×３の行列で表される。また、視点位置に対する物体の移動量は、物体の位置（座標値）と、視点位置の位置（座標値）と、視点位置の回転角とで生成される。ここで、回転角は、上述と同様に、２次元座標系が用いられる処理では２×２、３次元座標系が用いられる処理では３×３の行列で表される。尚、コントローラ２

２の操作に基づく物体の回転角、視点位置の回転角は、夫々テーブルとして保持されている。ＣＰＵ１は、コントローラ２２の操作に基いて、上記テーブルから、対応する物体や視点位置の回転角を読み出し、読み出した回転角を、視点位置に対する物体の回転量や、移動量を求めるために用いるのである。

【００３５】以上の説明から分かるように、表示エリア上のポリゴン頂点アドレスデータは、次のようにして求められる。即ち、コントローラ２２の操作に応じて、物体の回転角及び位置、並びに視点位置の回転角及び位置が、ＣＰＵ１によって求められる。次に、ＣＰＵ１により、物体の回転角と視点位置の回転角とに基いて、視点位置に対する物体の回転量が求められる。そして、ＣＰＵ１により、物体の位置と視点位置の位置及び回転角とに基いて、視点位置に対する物体の移動量が求められる。これら物体の回転量及び移動量データは、既に説明したように、３次元座標系が用いられて処理される場合においては、３×３の行列で表される。

【００３６】上記物体の回転量及び移動量データは、ポリゴンの絶対座標データと共に、グラフィックスデータ生成プロセッサ３に与えられる。グラフィックデータ生成プロセッサ３は、上記物体の回転量及び移動量データに基いて、ポリゴンの絶対座標データを、ポリゴン頂点アドレスデータに変換する。以上が、ポリゴン頂点アドレスデータが得られるまでの処理である。

【００３７】上記ポリゴン頂点アドレスデータは、バッファ１１の表示エリア上のアドレスを示す。描画処理プロセッサ１０は、バッファ１１の表示エリア上に、３若しくは４個のポリゴン頂点アドレスデータで示されるところの、三角形若しくは四角形１の範囲を設定し、当該範囲に、対応するテクスチャデータを書き込む。この処理は、一般に、「テクスチャの貼り付け」等と称されている。これによって、テレビジョンモニタ１２の表示面上には、多数のポリゴンにテクスチャデータの貼り付けられた物体が表示される。

【００３８】通常の２次元画像を描画するための描画命令は、頂点アドレスデータ、テクスチャアドレスデータ、カラーパレットアドレスデータ並びにテクスチャの輝度を示す輝度データとからなる。これらのデータの内、頂点アドレスデータは、グラフィックスデータ生成プロセッサ３が、ＣＰＵ１からの平面上における頂点座標データを、ＣＰＵ１からの移動量データに基いて、座標変換して得る座標データである。以下、描画処理については、「描画命令を発行する」等のように簡略化して記載する。

【００３９】音声処理プロセッサ１３は、記録媒体３０から読み出されたＡＤＰＣＭデータを、バッファ１４に記憶し、このバッファ１４に記憶されたＡＤＰＣＭデータを音源とする。そして、音声処理プロセッサ１３は、ＡＤＰＣＭデータを、例えば４４．１ＫＨｚの周波数のクロックで読み出す。そして、そして音声処理プロセッサ１３は、バッファ１４から読み出したＡＤＰＣＭデータに対し、ピッチの変換、ノイズの付加、エンベロープの設定、レベルの設定、リバーブの付加等の処理を施す。記録媒体３０から読み出される音声データがＰＣＭデータの場合においては、このＰＣＭデータは、ＣＰＵ１により、ＡＤＰＣＭデータに変換される。また、ＰＣＭデータに対するプログラムデータによる処理は、メインメモリ５上において直接行われる。メインメモリ５上において処理され、更に、ＡＤＰＣＭフォーマットのデータにエンコードされた音声データは、音声処理プロセッサ１３に供給されて上述した各種処理が施された後に、音声としてスピーカ１６から出力される。

【００４０】記録媒体ドライバ１９は、例えばハードディスクドライブ、光ディスクドライブ、フレキシブルディスクドライブ、シリコンディスクドライブ、カセット媒体読みとり機等である。記録媒体３０は、例えばハードディスク、光ディスク、フレキシブルディスク、半導体メモリ等である。記録媒体ドライバ１９は、記録媒体３０から画像、音声、ゲームプログラムデータを読み出し、読み出したデータを、デコーダ１７に供給する。デコーダ１７は、記録媒体ドライバ１９からの再生データに対し、ＥＣＣ（Ｅｒｒｏｒ　Ｃｏｒｒｅｃｔｉｏｎ　Ｃｏｄｅ）によるエラー訂正処理を施し、エラー訂正処理を施したデータを、メインメモリ５若しくは音声処理プロセッサ１３に供給する。

【００４１】メモリ２１は、例えば、ホルダ及びカード型のメモリからなる。カード型のメモリは、例えば終了時点の状態を保持する等のように、ゲームの各種パラメータを保持するためのものである。コントローラ２２は、左キーＬ、右キーＲ、上キーＵ、下キーＤからなる十字キーと、左ボタン２２Ｌ、右ボタン２２Ｒ、スタートボタン２２ａ、セレクトボタン２２ｂ、第１ボタン２２ｃ、第２ボタン２２ｄ、第３ボタン２２ｅ、第４ボタン２２ｆとからなる。十字キーは、ゲームプレーヤが、ＣＰＵ１に対し、上下左右を示すコマンドを与えるものである。スタートボタン２２ａは、ゲームプレーヤが、記録媒体３０からロードされるゲームプログラムデータの開始を、ＣＰＵ１に指示するためのものである。セレクトボタン２２ｂは、ゲームプレーヤが、記録媒体３０からメインメモリ５にロードされるゲームプログラムデータに関する各種選択を、ＣＰＵ１に指示するためのものである。尚、左キー２２Ｌ、右キー２２Ｒ、第１～第４ボタン２２ｃ、２２ｄ、２２ｅ、２２ｆの機能は、記録媒体３０からロードされるゲームプログラムデータによって異なる。

【００４２】〔動作〕電源スイッチ（図示せず）がオンにされ、ゲームシステムに電源が投入される。このとき、記録媒体３０が、記録媒体ドライバ１９に装填されていると、ＣＰＵ１が、ＲＯＭ６に記憶されているオペ

レーティングシステムに基いて、記録媒体ドライバ１９に対し、記録媒体３０からのゲームデータの読み出しを指示する。これにより、記録媒体ドライバ１９は、記録媒体３０から画像、音声及びゲームプログラムデータを読み出す。読み出された画像、音声及びゲームプログラムデータは、デコーダ１７に供給され、ここで、エラー訂正処理が施される。デコーダ１７においてエラー訂正処理の施された画像データは、バス２を介して伸張回路７に供給され、ここで、上述した伸張処理が施された後に、描画処理プロセッサ１０に供給され、この描画処理プロセッサ１０により、バッファ１１の非表示エリアに書き込まれる。

【００４３】デコーダ１７においてエラー訂正処理の施された音声データは、メインメモリ５若しくは音声処理プロセッサ１３に供給され、メインメモリ５若しくはバッファ１４に書き込まれる。また、デコーダ１７においてエラー訂正処理の施されたゲームプログラムデータは、メインメモリ５に供給され、このメインメモリ５に書き込まれる。以降、ＣＰＵ１は、メインメモリ５に記憶されているゲームプログラムデータ、並びにゲームプレーヤが、コントローラ２２を介して指示する内容に基いて、ゲームを進行する。即ち、ＣＰＵ１は、コントローラ２２を介してゲームプレーヤから指示される指示内容に基いて、適宜、画像処理の制御、音声処理の制御、内部処理の制御を行う。画像処理の制御とは、上述した回転量及び移動量データや絶対座標データのグラフィックスデータ生成プロセッサ３への供給、グラフィックスデータ生成プロセッサ３が求めたバッファ１１の表示エリア上のアドレスデータや輝度データを含む描画命令の発行等である。音声処理の制御とは、音声処理プロセッサ１３に対する音声出力コマンドの発行、レベル、リバーブ等の指定である。内部処理の制御とは、例えばコントローラ２２の操作に応じた演算等である。

【００４４】Ｂ．図１に示したＣＰＵ１の有する機能（図２）

【００４５】図２は、図１に示したＣＰＵ１の有する機能を示す説明図である。ＣＰＵ１は、図１に示した記録媒体３０から読み出され、メインメモリ５に記憶されたプログラムデータを読むことにより、図２に示す機能を持つ。この図２に示されているＣＰＵ１の機能は、ボタン操作検出手段１ａ、視点位置データ設定手段１ｂ、表示範囲情報抽出手段１ｃ、演算手段１ｄ、結果情報設定手段１ｅ、判断手段１ｆ、描画命令発行手段１ｇ、変数設定手段１ｈ、コマ数取得手段１ｉ、移動・回転量取得手段１ｊ、ポリゴン情報管理手段１ｋ、ガイド情報管理手段１ｍとで構成される。これらの手段は、夫々項目Ｅ～項目Ｇにおいて説明する制御の主体となる。

【００４６】Ｃ．矢印及び軌跡の表示（図３並びに図４Ｃ、Ｄ、Ｇ、及びＨ）

【００４７】図３Ａは、例えば縦４８０ピクセル、横６４０ピクセルで構成される表示面上における、矢印及び軌跡からなる投擲ガイドの表示の仕方を説明するための説明図、図３Ｂは、矢印の画像の絶対座標を示す説明図、図３Ｃは、図３Ｂに示した矢印の絶対座標と、移動量及び回転量データとに基いて、矢印を３次元座標上に設定したときの画像変形処理を概念的に説明するための説明図である。また、図４Ｃは、矢印絶対座標の一例を示す説明図、図４Ｄは、図４Ｃに示す矢印絶対座標データが、移動量及び回転量データに基いて、図１に示したグラフィックスデータ生成プロセッサ３によって変換された、変換後矢印アドレスの一例を示す説明図、図４Ｇは、図４Ｄに示される変換後矢印アドレスの内、矢印の尾に相当する変換後矢印後部アドレスの一例を示す説明図、図４Ｈは、図４Ｇに示される変換後矢印後部アドレスから得られる軌跡生成用アドレスの一例を示す説明図である。

【００４８】本形態においては、ゲームプレーヤの意図するところの投擲方向となる、ゲーム空間内の投擲体が被投擲物を投擲するタイミング、並びに投擲体の残りの回転数をゲームプレーヤに分かり易くガイドすることが目的とされる。この目的は、図４Ａに示されるように、投擲体が被投擲物を投擲する方向を示す矢印Ｎａと、この矢印Ｎａの軌跡Ｌｏが、テレビジョンモニタ１２の表示面上に表示されることにより達成される。しかも、矢印の示す方向は、その矢印と共に表示される選手等の投擲体の姿勢が示す被投擲物の投擲方向に一致される。つまり、表示面上に刻一刻と表示される投擲体の姿勢に応じた投擲方向を示す矢印が、投擲体と共に刻一刻と表示される。

【００４９】また、矢印Ｎａは、軌道ＯＲ上、且つ、矢印で示される方向に向かって、順に、表示される。このとき、現在表示されている矢印Ｎａの尾の部分の２つのアドレスデータａ１及びａ２と、１つ前に表示された矢印Ｎｂの尾の部分の２つのアドレスデータｂ２及びｂ１とが用いられて１つの画像が、生成、表示される。この画像が、矢印Ｎａの軌跡として用いられる軌跡画像Ｌｏである。この図３Ａに示されるように、軌跡画像Ｌｏは、矢印Ｎａの尾の部分のアドレスデータａ１と、矢印Ｎｂの尾の部分のアドレスデータｂ２とが結ばれ、矢印Ｎａの尾の部分のアドレスデータａ２と、矢印Ｎｂの尾の部分のアドレスデータｂ１とが結ばれることにより形成される、２つの三角形によって構成される。

【００５０】そして、更に、投擲体の回転数が、矢印Ｎａの色で表現される。例えば、回転数を示す値が最も少ないときの矢印Ｎａの色は、例えば紫色に設定され、回転数を示す値が大きくなるに従って矢印Ｎａの色は、順次暖色に設定され、回転数を示す値が最も大きいときの矢印Ｎａの色は、赤色に設定される。投擲ゲームが、砲丸投げの場合には、矢印は最初から赤色で表示される。砲丸投げ競技で主流となっている回転数は、０．５回転

だからである。また、投擲ゲームが、円盤投げの場合には、矢印は、1回転目では黄色、2回転目では赤色で夫々表示される。円盤投げ競技で主流となっている回転数は、2回転だからである。また、投擲ゲームが、ハンマー投げの場合には、矢印は、1回転目では紫色、2回転目では青色、3回転目では緑色、4回転目では黄色、5回転目では赤色で夫々表示される。ハンマー投げ競技で主流となっている回転数は、4回転であるが、本形態では5回転としているからである。尚、処理上においては、矢印の色は、演算手段1dが、投擲方位角度データAhを360で除算して得られる除算値（小数点以下切り捨て）に基いて、描画命令発行手段1gが、テーブルを参照することにより求められる。テーブルは、多数の除算値と、この多数の除算値に夫々対応するカラーパレットアドレスデータ若しくは色を示すデータとからなる。

【0051】また、本形態においては、上記最大の回転数を越えた場合には、失敗とされる。この失敗は、例えば、本物の競技におけるファールとされる。

【0052】以上のような投擲ガイド表示が、投擲体の姿勢に応じてリアルタイムで表示される。従って、ゲームプレーヤは、現時点で投擲体が被投擲物を投擲したときの、投擲方向及び残り回転数を知ることができる。従って、ゲームプレーヤは、上記投擲ガイド表示により、自分の意図する被投擲物の投擲方向及び意図するタイミングを認識し、この認識に基いてコントローラ22を操作して、投擲体に被投擲物を投擲させることができる。即ち、ユーザーインターフェースに優れた投擲ゲームを実現することができる。以下、より詳細に、上記ガイド表示を行うための処理について説明する。

【0053】矢印は、例えば図3Aに示すような形状である。そして、この矢印の絶対座標は、図3Aに示されるように、中心を“0”とした場合の、矢印を含む3次元平面上の矢印を囲む四辺形の角の座標データである。また、この矢印絶対座標データは、記録媒体30からメインメモリ5にロードされたデータである。この図3B及び図4Bに示される矢印絶対座標データの例では、（x1、y1、z1）が（8、0、10）、（x2、y2、z2）が（−8、0、10）、（x3、y3、z3）が（8、0、−10）、（x4、y4、z4）が（−8、0、−10）となっている。

【0054】図3Cは、グラフィックスデータ生成プロセッサ3による変換処理を概念的に示している。グラフィックスデータ生成プロセッサ3は、CPU1から、図3Bに示される矢印絶対座標データ、上述した移動量及び回転量データが与えられると、図3Cに示されるように、3次元座標上で矢印を設定する。説明の便宜上、カメラの位置と回転角は、常に投擲体と被投擲物とがいつも見える位置に固定されているものとする。

【0055】3次元座標上で設定された矢印を囲む四辺形の頂点のアドレスデータは、バッファ11の表示エリア上のアドレスデータに変換される。この変換後矢印アドレスの例は、図4Dに示されている。変換後矢印アドレスは、図4Dに示されるように、バッファメモリ11の表示エリア上におけるアドレスデータである。例えば図3Aに示されている現時点の矢印Naを囲む四辺形の角の、バッファ11の表示エリア上のアドレスa1、a2、a3及びa4は、夫々、（560、320）、（544、324）、（594、368）及び（560、394）となっている。また、現時点より1つ前の矢印Nbを囲む四辺形の角の、バッファ11の表示エリア上のアドレスb1、b2、b3及びb4は、夫々、（424、342）、（408、346）、（458、390）及び（424、416）となっている（図3A及び図4D参照）。

【0056】次に、軌跡画像Loについて説明する。矢印Naが現時点で表示され、矢印Nbが所定単位時間前に表示され、且つ、過去に表示された矢印が矢印Nbのみであるものとして説明する。

【0057】図3Aに示されるように、矢印Naが表示されているときには、メインメモリ5には、図4Dに示されるように、現時点で表示されている矢印Naの変換後矢印アドレスデータと、現時点より1つ前の時点で表示されていた矢印Nbの変換後アドレスデータが記憶されている。一方、メインメモリ5には、図4Gに示されるように、図4Dに示した変換後矢印アドレスの内、矢印の尾に相当するアドレスのみが、変換後矢印後部アドレスデータとして記憶される。図3Aを参照すると分かるように、現時点で表示されている矢印Na及び1フレーム前に表示された矢印Nbの尾部分のアドレスは、夫々、a1、a2、及びb1、b2である。そしてこれらの値は、この図3Aの下部に示されているように、（560、320）、（544、324）及び（424、342）、（408、346）である。これらの値が、図4Gに示されているように、メインメモリ5の他のエリアに、変換後矢印後部アドレスとして記憶される。

【0058】そして、図4Hに示されているように、矢印Na及びNbの両変換後矢印後部アドレスデータが、軌跡生成用アドレスデータとして、メインメモリ5に記憶された後に、描画処理プロセッサ10に供給される。描画処理プロセッサ10は、上記軌跡生成用アドレスデータを含む描画命令を受け取ると、このアドレスデータに基いて、バッファ11の表示エリア上に、矢印のテクスチャデータを、軌跡画像データとして書き込む。これにより、図3Aにおいて斜線を付して示すように、表示面上に、軌跡画像Loが表示される。この軌跡画像Loは、視覚的に、あたかも、矢印Naの軌跡のように見える。そして、矢印と軌跡が順次表示されることにより、矢印の軌道ORが視覚的に表現される。

【0059】この例では、現時点及び過去に表示された

矢印の数は２つであるから、表示される軌跡画像Ｌｏは１つであるが、この数が多数の場合は、表示される軌跡画像Ｌｏも多数となるので、表示面上に表示されている矢印に追従して表示される軌跡画像Ｌｏの長さが長くなる。

【００６０】軌跡画像Ｌｏの表示によって矢印の軌道ＯＲが表現されると、ゲームプレーヤは、予め矢印の表示される位置が分かるので、その分だけ、矢印で示される投擲方向に対する応答が向上する。つまり、何もないところに順次矢印が表示されるよりも、表現されている軌道ＯＲ上に矢印が表示される方が、矢印の表示位置を予測でき、これによって、投擲体に投擲させるべき矢印が表示されたときの操作反応を向上させることができる。

【００６１】次に、矢印Ｎａの色について説明する。既に説明したように、本形態では、投擲体の回転に応じて矢印Ｎａの色が変更される。これについては既に説明した。通常、寒色は、温度の低さ等のように小さいことを表現するために用いられる。また、暖色は、温度の高さ等のように大きいことを表現するために用いられる。本形態においては、回転数が多くなるのに従って、矢印Ｎａの色が、順次、寒色から暖色に変更される。従って、ゲームプレーヤは、残りの回転数を認識することができる。言い換えれば、ゲームプレーヤは、ゲーム空間内の投擲体に被投擲物を投擲させるタイミングを認識することができる。また、赤は、通常、「危険」や「限界」等を表現する色として用いられている。従って、回転数が上昇するにつれて矢印の色が赤に近づけられると、ゲームプレーヤは、残りの回転数が少なくなっていくことを確実に認識することができる。

【００６２】Ｄ．画像表示例（図５及び図６）

【００６３】図５Ａ～図５Ｄは、ハンマー投げゲームでの画面表示例を示す説明図である。この図５Ａ～図５Ｄは、ゲーム空間内の選手Ｍａが、ハンマーＢａを投げるための動作を開始してから、ハンマーＢａを投げ終えるまでの一連の多数のフレームの内の４つの画面表示例を抜き出して、時系列的に示している。時系列的順序は、図５Ａ、図５Ｂ、図５Ｃ、図５Ｄである。図５Ａはハンマーを投げるための動作を開始する時点の画面表示例を、図５Ｂはハンマーを投げるための回転動作を開始した時点の画面表示例を、図５Ｃはハンマーを投げる寸前の回転動作を行っている画面表示例を、図５Ｄはハンマーを投げた時点の画面表示例を示す。尚、便宜上、符号は、図５Ｂにのみ符号を付す。また、各画面表示例は、複数回ゲームが行われたときの全画面から選択されたものである。

【００６４】図５Ｂに示されているように、１つの画面は、背景の他、選手の画像Ｍａ、ハンマーの画像Ｂａ、矢印の画像Ｎａ、軌跡画像Ｌｏ、結果表示用の画像を表示するためのウインドウＷ及び角度表示用のガイド画像Ｇｉから構成される。また、背景画像中には、線ＬＬ及びＬＲが表示される。ウインドウＷの左端部分の上部には、ゲームプレーヤの数を示す文字（この例では、１Ｐ：一人）が表示され、その下部には、投擲エネルギーを示すバーグラフが表示され、更にその下部には、何人目のゲームプレーヤであるのかを示す文字（この例では、ＰＬＡＹＥＲ１）が表示される。また、ウインドウＷの中央には、夫々「１ＳＴ」、「２ＮＤ」、「３ＲＤ」の文字が表示される。これらは、夫々「１回目」、「２回目」、「３回目」の意味である。そして、上記「１ＳＴ」、「２ＮＤ」、「３ＲＤ」の文字の右隣には、飛距離若しくはファールを示す文字が表示される。図５Ａでは、１回目の投擲での飛距離が、「５９．３３Ｍ」（５９ｍ３３ｃｍを意味する）、２回目の投擲での飛距離が、「６８．３９Ｍ」（６８ｍ３９ｃｍを意味する）であることが示されている。尚、「１ＳＴ」の飛距離を示す文字の右側に示されている「×」、並びに「２ＮＤ」の飛距離を示す文字の右側に示されている「○」は、「１ＳＴ」での飛距離よりも「２ＮＤ」での飛距離が有効な値であることを示している。また、図５Ｂでは、１回目の投擲が、失敗、即ち、ファールであることが、「ＦＯＵＬ　×」の表示により示されている。

【００６５】また、図５Ｂに示されているように、軌跡画像Ｌｏは、矢印に近いほど大きく、遠いほど、小さくなるように表示される。

【００６６】Ｅ．メインルーチンによる制御動作（図７～図１０）

【００６７】図６～図９は、投擲ゲームのメインルーチンによる制御動作を説明するためのフローチャートである。投擲ゲームは、ゲームプレーヤによるコントローラ２２の操作に応じて、ゲーム空間内の投擲体により、被投擲物を、ゲーム空間内において投擲するといったものである。投擲競技としては、主に、投擲体が回転しまがら被投擲物を投擲するといった競技、例えば、砲丸投げ、円盤投げ、ハンマー投げが知られている。よって、図６～図９に示す投擲ゲームが、砲丸投げゲームの場合においては、「投擲体」は「選手」、「被投擲物」は「砲丸」と読み換え、円盤投げゲームの場合においては、「投擲体」は「選手」、「被投擲物」は「円盤」と読み換え、ハンマー投げゲームの場合においては、「投擲体」は「選手」、「被投擲物」は「ハンマー」と読み換えるものとする。

【００６８】尚、ステップＳ１のみ、図１に示したＲＯＭ６に記憶されているオペレーティングシステムによる制御動作である。他のステップは、記録媒体３０から読み出されたゲームプログラムデータによる制御動作である。また、ゲームプログラムデータによる制御の主体は、既に説明したように、図２に示したＣＰＵ１の機能としての各手段である。

【００６９】ステップＳ１では、オペレーティングシステムの命令により、記録媒体ドライバ１９が、記録媒体

３０から画像、音声及びゲームプログラムデータを読み出す。読み出されたデータの内、プログラムデータは、メインメモリ５に記憶される。これにより、ＣＰＵ１は、図２に示した機能を有する。尚、このとき、画像、即ち、テクスチャデータは、描画処理プロセッサ１０のバッファ１１の非表示エリアに記憶され、夫々テクスチャ番号が割り当てられるものとする。また、音声データは、音声処理プロセッサ１３のバッファ１４に記憶され、夫々音声番号データが割り当てられるものとする。通常、すべての画像及び音声データが、ステップＳ１においてバッファ１１及び１４に保持されることはないが、説明の便宜上、すべての画像及び音声データが、ステップＳ１においてロードされるものとする。ステップＳ２では、ボタン操作検出手段１ａが、コントローラ２２のスタートボタン２２ａが押された否かを判断し、「ＹＥＳ」であればステップＳ３に移行する。

【００７０】ステップＳ３では、描画命令発行手段１ｇが、セレクト画像の描画を示す描画命令を、図１に示した描画処理プロセッサ１０に対して発行する。描画処理プロセッサ１０は、上記描画命令に基いて、セレクト画像の画像データを、バッファ１１の表示面上に展開する。これにより、テレビジョンモニタ１２の表示面上には、セレクト画像が表示される。ステップＳ４では、ボタン操作手段１ａが、コントローラ２２のスタートボタン２２ａが押されたか否かを判断し「ＹＥＳ」であればステップＳ５に移行する。

【００７１】ステップＳ５では、ＣＰＵ１が、セレクトされたゲームにセットする。ここで、「セレクトされる」とは、ゲームプレーヤが、ステップＳ３で表示されたセレクト画像を参照して、十字キーを用いてゲームを選択し、この後に、スタートボタン２２ａを押すことを意味する。また、ここで「ゲーム」とは、ゲームそのものの他、例えば対戦型格闘ゲームにおけるキャラクタ等も含む。要するに、ゲームが実際に開始される前の選択事項である。説明の便宜上、このステップＳ５において、投擲ゲームが選択されたものとする。ステップＳ６では、描画命令発行手段１ｇが、セレクトされたゲームの初期画像の描画を示す描画命令を、描画処理プロセッサ１０に対して発行する。これにより、描画処理プロセッサ１０は、バッファ１１の表示エリア上に初期画像の画像データを書き込む。これにより、テレビジョンモニタ１２の表示面上には、初期画像が表示される。

【００７２】ステップＳ７では、変数設定手段１ｈが、メインメモリ５に保持しているフラグや変数を夫々リセットする。ステップＳ８では、ボタン操作検出手段１ａが、第１ボタン２２ｃが押されたか否かを判断し、「ＹＥＳ」であればステップＳ９に移行し、「ＮＯ」であればステップＳ１１に移行する。第１ボタン２２ｃの操作は、ゲーム空間内の投擲体の回転速度を制御するためのものである。

【００７３】ステップＳ９では、演算手段１ｄが、スピードデータＳｄに上昇時基準スピードデータｓを加算する。

【００７４】ステップＳ１０では、コマ数取得手段１ｉが、スピードデータＳｄの値に応じたコマ数データｆｄを求める。このコマ数データｆｄを、換算コマ数データと称する。換算コマ数データｆｄは、例えばテーブルを参照することにより求められる。テーブルは、多数のスピードデータＳｄと、これら多数のスピードデータＳｄについて夫々登録されている多数のコマ数データとからなり、記録媒体３０からメインメモリ５にロードされる。尚、スピードデータＳｄを用いて所定の演算を行うことにより、コマ数を求めるようにしても良い。

【００７５】ステップＳ１１では、演算手段１ｄが、スピードデータＳｄから下降時基準スピードデータｍを減算する。ステップＳ１２では、判断手段１ｆが、スピードデータＳｄの値がマイナスか否かを判断し、「ＹＥＳ」であればステップＳ１３に移行し、「ＮＯ」であればステップＳ１０に移行する。

【００７６】ステップＳ１３では、変数設定手段１ｈが、スピードデータＳｄに“０”を代入する。上記ステップＳ８で、ボタン操作検出手段１ａが、第１ボタン２２ｃが押されたか否かを判断し、第１ボタン２２ｃが押されていると判断したときに、ステップＳ９に移行してスピードデータＳｄを増加させ、第１ボタン２２ｃが押されていないものと判断したときに、ステップＳ１１に移行してスピードデータＳｄを減少させるのは、ゲームプレーヤによる単位時間あたりの第１ボタン２２ｃの押圧回数に応じて、スピードデータＳｄの値を設定するためである。つまり、ゲームプレーヤが単位時間内に第１ボタン２２ｃを押す回数が多いと、ゲーム空間内の投擲体の回転速度、即ち、投擲エネルギーが上昇する。これに対し、ゲームプレーヤが単位時間内に第１ボタン２２ｃを押す回数が低いと、ゲーム空間内の投擲体の回転速度、即ち、投擲エネルギーが下降する。フローチャート中にステップとして示されていないが、上記投擲エネルギーの表示は、図５Ａ～図５Ｄに示したバーグラフの変化により表される。

【００７７】ステップＳ１４では、演算手段１ｄが、コマ数データＦＤに、換算コマ数データｆｄを、加算する。ステップＳ１５では、コマ数取得手段１ｉが、ステップＳ１４において求められたコマ数データＦＤに対応する投擲方位角度データＡｈを、メインメモリ５に保持されているテーブルＴＢＬから、読み出す。ここで、テーブルＴＢＬは、図４Ａに示されているように、多数のコマ数データＦＤと、この多数のコマ数データＦＤについて夫々登録されている多数の投擲方位角度データＡｈとからなる。

【００７８】ステップＳ１６では、変数設定手段１ｈが、投擲体のポリゴンの絶対座標データのメインメモリ

５上のアドレスデータ、コマ数データＦＤの値に応じた移動量データ及び回転量データを、テーブルから読み出し、上記アドレスデータを、アドレス変数ＡＤＤに、上記移動量データを、移動量変数ＭＯに、上記回転量データを、回転量変数ＲＯに夫々代入する。ここで、上記アドレスデータ、移動量データ並びに回転量データは、テーブルを参照することにより得られる。このテーブルは、最小値から最大値までのコマ数と、これらのコマ数の値に応じて夫々登録されている多数のアドレスデータ、移動量データ並びに回転量データからなる。

【００７９】ステップＳ１００では、ポリゴン画像表示処理が行われる。このポリゴン画像表示ルーチンＳ１００については、後に詳述する。ステップＳ２００では、投擲ガイド表示処理が行われる。この投擲ガイド表示ルーチンＳ２００については、後に詳述する。

【００８０】ステップＳ１７では、判断手段１ｆが、投擲方位角度データＡｈの値が、投擲方位角度の最小値データＡｈｍｉｎの値以下か否かを判断し、「ＹＥＳ」であればステップＳ１８に移行し、「ＮＯ」であればステップＳ２２に移行する。ここで、投擲方向角度の最小値データＡｈｍｉｎは、図４Ａに示すように、最も小さい角度、且つ、被投擲物が余裕をもって非有効エリアに落下する角度である。ここで、非有効エリアは、図４Ｂ、図５Ａ～図５Ｄに夫々示されているように、例えば線ＬＬ及びＬＲ間で示される範囲外のエリアである。

【００８１】本形態において決められている投擲時の最大回転数は、砲丸投げで“０．５”、円盤投げで“２”、ハンマー投げで“５”となっている。そこで、本形態においては、上記最大回転数に応じて、コマ数の最大値が決められている。例えば、或競技で設定されている回転数が４回で、その最大コマ数の値が、“２４０”に設定されているものとすると、回転数が２回の競技における最大コマ数の値は、“１２０”となる。また、上記陸上競技において「失敗」は、「ファール」の意味である。

【００８２】ステップＳ１７において、判断手段１ｆが、「ＮＯ」と判断した場合には、ステップＳ１８に移行し、このステップＳ１８では結果情報設定手段１ｅが、描画処理プロセッサ１０に対し、“ＦＯＵＬ　×”を示す文字データを供給する。これにより、ウインドウＷの中央部分には、図５Ｂに示すように、「ＦＯＵＬ×」を示す文字が、画像として表示される。ステップＳ３００では、失敗用投擲体画像表示処理が行われる。この失敗用投擲体画像表示ルーチンＳ３００は、後述するステップＳ３２で行われるような設定処理及びステップＳ１００で行われる処理からなる。表示面上では、例えば投擲体が、がっくりと倒れ込む等のように、失敗時における投擲体の動きが表現される。

【００８３】ステップＳ１９では、判断手段１ｆが、ボタン操作検出手段１ａからの、押されたボタンを示すデータの有無により、何れかのボタンが押されたか否かを判断し、「ＹＥＳ」であればステップＳ２０に移行し、「ＮＯ」であれば再びステップＳ３００に移行する。ステップＳ２０では、演算手段１ｄが、投擲回数データＴｈに“１”を加算する。

【００８４】ステップＳ２１では、判断手段１ｆが、投擲回数データＴｈが、投擲回数データの最大値Ｔｈｍａｘよりも大きいか否かを判断し、「ＹＥＳ」であれば再びステップＳ３に移行し、「ＮＯ」であれば再びステップＳ８に移行する。ステップＳ２２では、ボタン操作検出手段１ａが、第２ボタン２２ｄが押されたか否かを判断し、「ＹＥＳ」であればステップＳ２３に移行し、「ＮＯ」であれば再びステップＳ８に移行する。ここで、第２ボタン２２ｄは、被投擲物の投擲上下角度データＡｖと、投擲タイミングを決定する。この形態では、第２ボタン２２ｄが押されている間に、投擲上下角度データＡｖの値が順次インクリメントされる。尚、上記投擲上下角度データＡｖの値は、図５Ａ～図５Ｄに示すように、ガイドＧｉによってリアルタイムで表示される。

【００８５】ステップＳ２３では、演算手段１ｄが、投擲上下角度データＡｖに基準角度データｚを加算する。ステップＳ２４では、判断手段１ｆが、投擲上下角度データＡｖの値が、投擲上下角度データＡｖの最大値データＡｖｍａｘの値よりも大きいか否かを判断し、「ＹＥＳ」であればステップＳ２５に移行し、「ＮＯ」であればステップＳ２６に移行する。

【００８６】ステップＳ２５では、変数設定手段１ｈが、投擲上下角度データＡｖに、投擲上下角度データＡｖの最大値データＡｖｍａｘを、代入する。ステップＳ２６では、ボタン操作検出手段１ａが、第２ボタン２２ｄが離されたか否かを判断し、「ＹＥＳ」であればステップＳ２７に移行し、「ＮＯ」であれば再びステップＳ２３に移行する。第２ボタン２２ｄが離されたときに、その時点の投擲上下角度データＡｖの値で、ゲーム空間内の投擲体により被投擲物が投擲される。

【００８７】ステップＳ２７では、変数設定手段１ｈが、スピードデータＳｄの値と、投擲方位角度データＡｈと、投擲上下角度データＡｖの値とに基いて、被投擲物の速度ベクトルデータを、初期化する。ここで、速度ベクトルデータは、３次元座標上での位置を示し、（ｘ、ｙ、ｚ）からなる。ここで、「初期化」とは、上記３つの値によって決まる速度ベクトルデータを設定することを意味する。ステップＳ２８では、演算手段１ｄが、ステップＳ２７で初期化された速度ベクトルデータに、重力加速度ベクトルデータを、加算する。ここで、重力加速度ベクトルデータは、上記速度ベクトルデータが示すところの、３次元座標上における位置（ｘ、ｙ、ｚ）を変えるための定数である。

【００８８】ステップＳ２９では、演算手段１ｄが、非投擲物の位置データ（ｘ、ｙ、ｚ）に、速度ベクトルデ

ータ（ｘ、ｙ、ｚ）を、加算する。ここで、非投擲物の位置データは、３次元座標上における被投擲物の位置を示し、（ｘ、ｙ、ｚ）からなる。ステップＳ３０では、判断手段１ｆが、被投擲物の高さが“０”か否かを判断し、「ＹＥＳ」であればステップＳ３１に移行し、「ＮＯ」であればステップＳ３２に移行する。ここで、高さとは、例えばフレーム毎の被投擲物の表示面上における高さを意味する。

【００８９】ステップＳ３１では、変数設定手段１ｈが、被投擲物の高さを、“０”にする。ステップＳ３２では、変数設定手段１ｈが、被投擲物のポリゴンの絶対座標データのメインメモリ５上のアドレスデータをアドレス変数ＡＤＤに、上記移動量データを移動量変数ＭＯに、上記回転量データを回転量変数ＲＯに夫々代入する。ここで、上記移動量データ並びに回転量データは、テーブルを参照することにより得られる。このテーブルは、最小値から最大値までの被投擲物の位置データと、これらの位置データの値に応じて夫々登録されている多数の移動量データ並びに回転量データからなる。勿論、上記移動量データ及び回転量データは、上記位置データに基づき、随時、演算によって求めても良い。

【００９０】ステップＳ１００では、ポリゴン画像表示処理が行われる。ステップＳ４００では、投擲後投擲体画像表示処理が行われる。この投擲体画像表示ルーチンＳ４００は、ステップＳ３２で行われる処理及びステップＳ１００で行われる処理からなるルーチンである。表示面上においては、投擲体が被投擲物を投擲した後の動きが表現される。

【００９１】ステップＳ３３では、演算手段１ｄが、飛距離を求める。そして、結果情報設定手段１ｅが、飛距離を示す文字データを、描画処理プロセッサ１０に供給する。ステップＳ３４では、描画命令発行手段１ｇが、結果表示用の画像の表示を示す描画命令を、描画処理プロセッサ１０に供給する。これによって、テレビジョンモニタ１２の表示面上には、図５Ａ～図５Ｄに示したウインドウＷの中央部分に、被投擲物の飛距離を示す文字が、画像として表示される。

【００９２】ステップＳ３５では、判断手段１ｆが、ボタン操作検出手段１ａからの操作ボタンを示すデータの有無に基いて、何れかのボタンが押されたか否かを判断し、「ＹＥＳ」であればステップＳ３６に移行する。ステップＳ３６では、描画命令発行手段１ｇが、成功時の画像の描画を示す描画命令を、描画処理プロセッサ１０に対し発行する。これにより、描画命令プロセッサ１０は、成功時の画像データを、バッファ１１の表示エリア上に書き込む。従って、テレビジョンモニタ１２の表示面上には、成功時の画像が表示される。ここで、成功時の画像とは、例えば、「ＧＯＯＤ」等の文字の画像等であり、例えば、図５Ａ～図５Ｄに示したウインドウＷの中央部分に表示される。

【００９３】ステップＳ５００では、成功時投擲体画像表示処理が行われる。この成功時投擲体画像表示ルーチンＳ５００は、ステップＳ３２で行われる処理及びステップＳ１００で行われる処理からなるルーチンである。表示面上では、例えば投擲体が飛び跳ねる等のように、成功時における投擲体の動きが表現される。ステップＳ６００では、リプレー画像表示処理が行われる。このリプレー画像表示処理は、ゲームプレーヤによるコントローラ２２の操作情報が用いられる。即ち、ゲームプレーヤによるコントローラ２２の操作情報は、メインメモリ５に逐一記憶される。そして、このリプレー画像表示処理において、上記操作情報に基いて順次処理が行われる。これにより、ゲームプレーヤが行った操作に基いた画像表示状態を再現することができる。

【００９４】ステップＳ３７では、判断手段１ｆが、ボタン操作検出手段１ａからの操作内容を示すデータの有無により、何れかのボタンが押されたか否かを判断し、「ＹＥＳ」であれば再びステップＳ２０に移行する。

【００９５】Ｆ．ポリゴン画像表示ルーチンＳ１００による制御動作（図１１）

【００９６】図１０は、ポリゴン画像表示ルーチンＳ１００による制御動作を説明するためのフローチャートである。このポリゴン画像表示ルーチンＳ１００では、投擲体や被投擲物のポリゴン表示処理が行われる。１つの投擲体や被投擲物は、多数のポリゴンで構成される。図４Ｅに示されるように、ポリゴンは、多数のポリゴン頂点の絶対座標データ（ｘ、ｙ、ｚ）としてメインメモリ５に記憶されている。そして、これらの絶対座標データ（ｘ、ｙ、ｚ）は、グラフィックスデータ生成プロセッサ３により、上述した移動量データ及び回転量データに基いて、図４Ｆに示される２次元平面上の変換後ポリゴンアドレスデータ（ｘ、ｙ）に変換される。この図４Ｆに示される変換後ポリゴンアドレスデータ（ｘ、ｙ）は、テクスチャアドレスデータやカラーパレットアドレスデータと共に、描画命令として、描画処理プロセッサ１０に供給される。描画処理プロセッサ１０は、描画命令に基いて、バッファ１１の表示エリア上に、上記変換後ポリゴンアドレスデータに基いて、テクスチャデータを書き込む。これにより、テレビジョンモニタ１２の表示面上には、多数のポリゴンからなる投擲体や被投擲物が表示される。

【００９７】ステップＳ１０１では、ポリゴン情報管理手段１ｋが、メインメモリ５から、アドレス変数ＡＤＤの値が示すメインメモリ５上のポリゴンの頂点の絶対座標データ（ｘ、ｙ、ｚ）を読み出す。ステップＳ１０２では、ポリゴン情報管理手段１ｋが、ポリゴンの頂点の絶対座標、移動量変数ＭＯに代入されている移動量、回転量変数ＲＯに代入されている回転量、光線のベクトル、ポリゴンの法線のデータを、グラフィックスデータ生成プロセッサ３に夫々供給する。グラフィックスデー

タ生成プロセッサ３は、上記各データに基いて、変換後ポリゴンアドレスデータ（ｘ、ｙ）を及び輝度データを夫々求め、これらのデータを、ポリゴン情報管理手段１ｋに夫々供給する。ステップＳ１０３では、ポリゴン情報管理手段１ｋが、グラフィックスデータ生成プロセッサ３からの、変換後ポリゴンアドレスデータ（ｘ、ｙ）及び輝度データを、夫々メインメモリ５に書き込む。

【００９８】ステップＳ１０４では、判断手段１ｆが、ポリゴンの頂点の絶対座標データを全て変換後ポリゴンアドレスデータに変換したか否かを判断し、「ＹＥＳ」であればステップＳ１０５に移行し、「ＮＯ」であれば再びステップＳ１０２に移行する。ステップＳ１０５では、描画命令発行手段１ｇが、変換後アドレスデータ（ｘ、ｙ）及び輝度データをメインメモリ５から読み出し、当該変換後アドレスデータ（ｘ、ｙ）及び輝度データを、テクスチャアドレスデータやカラーパレットアドレスデータと共に、描画命令として、描画処理プロセッサ１０に供給する。これにより、描画処理プロセッサ１０は、上記変換後アドレスデータ（ｘ、ｙ）に基いて、投擲体のテクスチャデータを、バッファ１１の表示エリア上に書き込む。従って、テレビジョンモニタ１２の表示面上には、多数のポリゴンからなる投擲体や被投擲物等の画像データが表示される。

【００９９】ステップＳ１０６では、判断手段１ｆが、全てのデータを転送したか否かを判断し、「ＹＥＳ」であればこのポリゴン画像表示ルーチンＳ１００を抜け、「ＮＯ」であれば再びステップＳ１０５に移行する。

【０１００】Ｇ．投擲ガイド表示ルーチンＳ２００による制御動作（図１２～図１４）

【０１０１】図１１～図１３は、投擲ガイド表示ルーチンによる制御動作を説明するためのフローチャートである。投擲ガイドは、矢印の画像と、この矢印の軌跡を示す軌跡画像とからなる。

【０１０２】ステップＳ２０１では、ガイド情報管理手段１ｍが、矢印を囲む四辺形の頂点の絶対座標データを、メインメモリ５から読み出す。ステップＳ２０２では、移動・回転量取得手段１ｊが、投擲方位角度データＡｈの値と、図３Ａに示した回転中心Ｏからの距離データに応じて、矢印を囲む四辺形の３次元座標上での移動量データ及び回転量データを得る。ここで、上記距離データは、固定値である。

【０１０３】ステップＳ２０３では、ガイド情報管理手段１ｍが、矢印の絶対座標データと、移動量データと、回転量データを、グラフィックスデータ生成プロセッサ３に供給する。これにより、グラフィックスデータ生成プロセッサ３は、矢印の絶対座標データを、移動量データ及び回転量データに基いて３次元座標上で変換し、この変換により得られた座標データから、２次元上の変換後アドレスデータ（ｘ、ｙ）を得、この変換後アドレスデータ（ｘ、ｙ）を、ガイド情報管理手段１ｍに供給する。ステップＳ２０４では、ガイド情報管理手段１ｍが、グラフィックスデータ生成プロセッサ３からの変換後アドレスデータ（ｘ、ｙ）を、メインメモリ５に書き込む。

【０１０４】ステップＳ２０５では、描画命令発行手段１ｇが、変換後アドレスデータ（ｘ、ｙ）をメインメモリ５から読み出し、当該変換後アドレスデータ（ｘ、ｙ）を、テクスチャアドレスデータやカラーパレットアドレスデータと共に、描画命令として、描画処理プロセッサ１０に供給する。尚、描画命令発行手段１ｇは、回転数に応じた色で矢印を表示するためのカラーパレットアドレスデータを、演算手段１ｄが行う演算値に基いて取得する。ここで演算値は、投擲方位角度データＡｈを３６０（度）で除算することにより求められる。これにより、描画処理プロセッサ１０は、変換後アドレスデータ（ｘ、ｙ）に基いて、バッファ１１の表示エリア上に、矢印のテクスチャデータを、カラーパレットにより指定される色で書き込む。尚、色の決定は、テーブルの参照により行われる。テーブルは、演算手段１ｄにより求められる最小値から最大値までの除算値と、これらの乗算値について夫々登録されているところの、色を指定するためのデータ若しくはカラーパレットアドレスデータとからなる。

【０１０５】ステップＳ２０６では、判断手段１ｆが、アドレスポインタＰの値が先頭アドレスＰｓｔａｒｔか否かを判断し、「ＹＥＳ」であればステップＳ２０７に移行し、「ＮＯ」であればステップＳ２０９に移行する。ここで、Ｐｓｔａｒｔは、変換後アドレスデータが記憶されるエリアにおける先頭アドレスである。ステップＳ２０７では、演算手段１ｄが、アドレスポインタＰに基準アドレス数ｋを加算する。ここで、基準アドレス数ｋの値は、２つのアドレスデータを記憶するのに必要な容量値である。

【０１０６】ステップＳ２０８では、ガイド情報管理手段１ｍが、メインメモリ５上のアドレスポインタＰの値が示すエリアに、矢印の尾に対応する平面の変換後アドレスデータ（ｘ、ｙ）を記憶する。そしてこの投擲ガイド表示ルーチンＳ２００を抜ける。ステップＳ２０９では、演算手段１ｄが、アドレスポインタＰの示す値が、先頭アドレスＰｓｔａｒｔと、変換後アドレスデータの最大記憶数ｎｍａｘとの加算結果から、基準アドレス数ｋを減じた減算結果の示す値よりも小さいか否かを判断し、「ＹＥＳ」であればステップＳ２１０に移行し、「ＮＯ」であればステップＳ２１１に移行する。ここで、変換後アドレスデータの最大記憶数ｎｍａｘの最小単位はｋである。また、このステップＳ２０９における処理は、アドレスポインタＰの値にリミッタをかけることにより、多数の軌跡画像Ｌｏにより表示される軌跡の形を、円弧とするために行われる。もしも、この処理によって上記アドレスポインタＰの値にリミッタがかけら

れない場合には、多数の軌跡画像Ｌｏにより表示される軌跡の形が、円になってしまう。

【０１０７】ステップＳ２１０では、演算手段１ｄが、アドレスポインタＰに、基準アドレス数ｋを加算する。ステップＳ２１１では、後アドレスＡＤに、アドレスポインタＰの値を代入する。

【０１０８】ステップＳ２１２では、ガイド情報管理手段１ｍが、前アドレスａｄにアドレスポインタＰの値よりも“ｋ”だけ少ない値を代入する。ここで、後アドレスＡＤの示す値と、前アドレスａｄの示す値との関係は、次に示す通りである。

ＡＤ＜ａｄ

【０１０９】ステップＳ２１３では、ガイド情報管理手段１ｍが、メインメモリ５上の前アドレスａｄの値が示すエリアに記憶されている変換後アドレスデータ（ｘ、ｙ）を、メインメモリ５上の後アドレスＡＤの値が示すエリアに記憶する。ステップＳ２１４では、演算手段１ｄが、前アドレスａｄから、基準アドレス数ｋを減算する。

【０１１０】ステップＳ２１５では、演算手段１ｄが、後アドレスＡＤから、基準アドレス数ｋを減算する。ステップＳ２１６では、判断手段１ｆが、前アドレスａｄの示す値が、先頭アドレスＰｓｔａｒｔに基準アドレス数ｋを加算した加算結果の示す値以下か否かを判断し、「ＹＥＳ」であればステップＳ２１７に移行し、「ＮＯ」であれば再びステップＳ２１３に移行する。

【０１１１】上記ステップＳ２０９～ステップＳ２１６までの処理は、メインメモリ５上において、矢印の尾に対応する変換後アドレスデータ（ｘ、ｙ）を、順次、大きな値のアドレスに対応するエリアに移す処理であり、いわばシフトレジスタとしての処理である。ステップＳ２１１において後アドレスＡＤにアドレスポインタＰを代入し、ステップＳ２１２において前アドレスａｄにアドレスポインタＰよりｋだけ少ないデータを代入するのは、ステップＳ２１３において、前アドレスａｄの値が示すエリアに記憶されている変換後アドレスデータ（ｘ、ｙ）を、後アドレスＡＤの値が示すエリアに記憶するためである。つまり、この処理が、いわばシフトレジスタとしての処理である。また、ステップＳ２１４において後アドレスＡＤから基準アドレス数ｋを減じ、ステップＳ２１５において前アドレスａｄから基準アドレス数ｋを減じ、ステップＳ２１６において前アドレスａｄの示す値が、先頭アドレスＰｓｔａｒｔに基準アドレス数ｋを加算した加算結果の示す値以下か否かを判断しているのは、前アドレスａｄに記憶されている変換後アドレスデータ（ｘ、ｙ）を記憶するエリアが有るか否かを判断するためである。つまり、前アドレスａｄの値が上記加算結果の示す値以下になるということは、後アドレスＡＤに対応する記憶エリアがないことになるからである。

【０１１２】ステップＳ２１７では、ガイド情報管理手段１ｍが、矢印の尾に対応する平面の頂点の変換後アドレスデータ（ｘ、ｙ）を、メインメモリ５の後アドレスＡＤの示すエリアに記憶する。

【０１１３】ステップＳ２１８では、変数設定手段１ｈが、第１アドレスＡＤ１に、先頭アドレスＰｓｔａｒｔと基準アドレス数ｋとの加算結果を、代入する。ステップＳ２１９では、変数設定手段１ｈが、第２アドレスＡＤ２に、先頭アドレスＰｓｔａｒｔと基準アドレス数の倍２ｋとの加算結果を代入する。

【０１１４】ステップＳ２２０では、ガイド情報管理手段１ｍが、第１、第２アドレスＡＤ１、ＡＤ２が示すメインメモリ５上の変換後アドレスデータ（ｘ、ｙ）を読み出し、当該変換後アドレスデータ（ｘ、ｙ）を、テクスチャアドレスデータやカラーパレットデータと共に、描画命令として、描画処理プロセッサ１０に供給する。ここで、第１アドレスＡＤ１が示すメインメモリ５上のエリアに記憶されているデータは、現時点に表示されている矢印の尾に相当する変換後アドレスデータである。また、第２アドレスＡＤ２が示すメインメモリ５上のエリアに記憶されているデータは、現時点よりも１単位期間だけ前に表示された矢印の尾に相当する変換後アドレスデータである。つまり、描画処理プロセッサ１０には、現時点及び現時点より１単位期間だけ前に表示された矢印の尾に相当する変換後アドレスデータ（ｘ、ｙ）が、１つの画像のアドレスデータとして供給される。従って、描画処理プロセッサ１０は、上記４つの変換後アドレスデータ（ｘ、ｙ）に基いて、バッファ１１の表示エリア上に、矢印のテクスチャデータを、軌跡画像Ｌｏのテクスチャデータとして書き込む。このときの軌跡画像Ｌｏの色は、現時点で表示されている矢印Ｎａの色と同じである。ステップＳ２２１では、演算手段１ｄが、第１アドレスＡＤ１に基準アドレス数ｋを、加算する。

【０１１５】ステップＳ２２２では、演算手段１Ｄが、第２アドレスＡＤ２に基準アドレス数ｋを、加算する。ステップＳ２２３では、判断手段１ｆが、第１アドレスＡＤ１の値が、アドレスポインタＰの値以上か否かを判断し、「ＹＥＳ」であればこの投擲ガイド表示ルーチンＳ２００を抜け、「ＮＯ」であれば再びステップＳ２２０に移行する。

【０１１６】上記ステップＳ２１７～ステップＳ２２３までの処理は、メインメモリ５に記憶されている変換後アドレスデータ（ｘ、ｙ）を、２エリア分ずつ、１つの四辺形の頂点のアドレスデータとして、描画処理プロセッサ１０に供給するための処理である。ステップＳ２１８において、第１アドレスＡＤ１に先頭アドレスＰｓｔａｒｔと基準アドレス数ｋとの加算結果を代入し、ステップＳ２１９において第２アドレスＡＤ２に先頭アドレスＰｓｔａｒｔと基準アドレス数の倍２ｋとの加算結果を代入しているのは、１つの矢印の尾に相当する変換後

アドレスデータ（ｘ、ｙ）と、上記矢印よりも１つ前に表示された矢印の尾に相当する変換後アドレスデータ（ｘ、ｙ）とを、１つの画像の変換後アドレスデータとして、描画処理プロセッサ１０に供給するためである。

【０１１７】また、ステップＳ２２１において第１アドレスＡＤ１に基準アドレス数ｋを加算し、ステップＳ２２２において第２アドレスＡＤ２に基準アドレス数ｋを加算し、ステップＳ２２３において第１アドレスＡＤ１の示す値がアドレスポインタＰ以上か否かを判断しているのは、第１アドレスＡＤ１に記憶されている変換後アドレスデータ（ｘ、ｙ）と対となる、第２アドレスＡＤ２に対応する変換後アドレスデータ（ｘ、ｙ）が有るか否かを判断するためである。つまり、第１アドレスＡＤ１の示す値が、アドレスポインタＰが示す値以上になるということは、第２アドレスＡＤ２に対応する変換後アドレスデータがないことになるからである。

【０１１８】〔実施の形態における効果〕以上説明したように、本形態においては、コントローラ２２の操作に応じた速度で投擲体を回転させると共に、コントローラ２２の操作により、投擲体に被投擲物を投擲させるようにした投擲ゲームにおいて、矢印及び軌跡の画像を投擲方向に基いてリアルタイムで表示するようにしたので、ゲームプレーヤは、リアルタイムで、被投擲物の投擲方向を認識でき、自分の意図した方向に、被投擲物を投擲させることができる。更に、投擲体の回転数の値が大きくなるに従って、矢印の色を寒色から暖色に順次変更するようにしたので、ゲームプレーヤは、残りの回転数を認識することができ、よって、リアルタイムで、被投擲物の投擲タイミングを認識することができる。

【０１１９】〔変形例１〕上記実施の形態においては、投擲体の回転数に応じて矢印の色を可変する場合について説明したが、投擲方位角度データＡｈの累積値に基いて矢印の色を可変するようにしても良い。つまり、より細かく矢印の色分けを行うようにする。この場合においても、上記累積値の値が小さい値から大きい値になるに従って、矢印の色を、寒色から暖色にする。以上のようにすれば、色数が増加するので、より細かく投擲タイミングをゲームプレーヤに報知することができるという効果がある。

【０１２０】〔変形例２〕上記実施の形態においては、軌跡画像の色は、矢印の色と同じ色にすることとしたが、例えば、白色としても良い。以上のようにすれば、ゲームプレーヤは、矢印と、矢印の軌跡を視覚的に区別することができ、投擲方向を確実に把握することができるという効果がある。

【０１２１】〔変形例３〕上記実施の形態においては、回転数に応じて矢印の色を変えるようにした場合について説明したが、投擲エネルギーの値が大きくなるにつれて、矢印の色を、寒色から暖色にするようにしても良い。以上のようにすれば、被投擲物の飛距離が伸びるタイミングを、ゲームプレーヤに対し、分かり易くガイドすることができる。

【０１２２】

【発明の効果】上述せる本発明によれば、少なくとも、投擲体を示す画像情報と、当該投擲体により視覚的に投擲される被投擲物を示す画像とが表示手段の表示面上に表示され、操作手段の操作に基いて、上記投擲体が視覚的に上記被投擲物を投擲するために動作させられると共に、上記操作手段の操作に基いて、上記被投擲物が上記表示手段の表示面上において上記投擲体により視覚的に投擲される、投擲ゲームにおいて、上記投擲体が上記被投擲物を投擲するための動作によって随時変化する上記被投擲物の投擲方向を示す投擲ガイド画像を表示するようにしたので、ゲームプレーヤに対して、被投擲物を投擲する方向を認識させることができるという効果がある。

【０１２３】また、上記発明において、上記投擲体が上記被投擲物を投擲するための動作は、弧を描く動作若しくは略回転動作である。従って、ゲームプレーヤは、投擲体が、弧を描く動作若しくは回転動作を行っているときであっても、どのタイミングで被投擲物を投擲体に投擲させれば、意図した方向に被投擲物が投擲されるのかを、認識することができる。

【０１２４】また、上記発明において、上記投擲ガイド画像は、上記投擲体の動作に応じた軌道を描くように順次表示される。従って、ゲームプレーヤは、投擲体の動作による軌道を、投擲ガイド画像を見ることだけで認識することができ、よって、投擲ガイド画像を参照することに集中し、より被投擲物の被距離を延ばすことができるという効果がある。

【０１２５】また、上記発明において、上記投擲ガイド画像の軌跡を示す軌跡画像が、上記投擲体の動作に応じた軌道を描くように順次表示される。よって、投擲ガイド画像が、表示面上のどの位置に出現するのかを、ゲームプレーヤに予測させることができ、これによって、ゲームプレーヤに、自分の意図した投擲タイミングをつかみ易くさせることができるという効果がある。

【０１２６】また、上記発明において、上記軌跡画像は、１つの投擲ガイド画像の座標情報と、当該投擲ガイド画像よりも１つ前に表示された投擲ガイド画像の座標情報に基いて生成される。よって、表示される投擲ガイド画像の位置に応じた正確な軌道を表現できると共に、投擲ガイド画像の座標情報を再利用することにより、処理速度を向上させることができるという効果がある。

【０１２７】また、上記発明において、上記投擲体及び上記投擲ガイド画像は、上記表示手段の表示面上において、疑似３次元表示される。よって、投擲体と投擲ガイド画像との視覚的マッチングをとると共に、ゲームプレーヤに操作し易い環境を与えることができるという効果がある。

【０１２８】また、上記発明において、上記投擲ガイド画像の形状は、少なくとも矢印形状若しくはこれに類する、視覚的方向指示機能を有する形状である。よって、ゲームプレーヤに対し、投擲方向をより明確に示すことができ、これにより、ゲームプレーヤに対し、より操作し易い環境を与えることができるという効果がある。

【０１２９】また、上記発明において、上記投擲体の回転数に応じて投擲ガイド画像の色が可変される。従って、ゲームプレーヤは、残りの回転数を認識でき、よって投擲を行うタイミングを分かり易く示すことができるといった効果がある。

【０１３０】また、上記発明において、上記投擲ガイド画像の色は、投擲体の回転数の値が小さい値から大きい値に変化されるのに従って、寒色から暖色に変化される。よって、ゲームプレーヤにより操作し易い環境を与えることができるという効果がある。

【図面の簡単な説明】

【図１】本発明の一実施の形態を示すゲームシステムの構成図である。

【図２】図１に示したＣＰＵ１が有する機能を示す機能ブロック図である。

【図３】矢印及び軌跡を説明するための説明図である。

〔図３Ａ〕　軌道と矢印の関係及び軌跡の一例を示す説明図である。

〔図３Ｂ〕　矢印の絶対座標の一例を示す説明図である。

〔図３Ｃ〕　矢印の絶対座標から疑似３次元表示する際の概念を説明するための説明図である。

【図４】テーブル及びメインメモリ上における画像情報の状態を示す説明図である。

〔図４Ａ〕　コマ数とこれに対応する矢印角度データからなるテーブルを示す説明図である。

〔図４Ｂ〕　投擲フィールド及び投擲方位角度データの最小値を示す説明図である。

〔図４Ｃ〕　矢印絶対座標の一例を示す説明図である。

〔図４Ｄ〕　図３Ｃに示す絶対座標から変換された変換後矢印アドレスの一例を示す説明図である。

〔図４Ｅ〕　ポリゴン絶対座標の一例を示す説明図である。

〔図４Ｆ〕　図３Ｅに示す絶対座標から変換された変換後ポリゴンアドレスの一例を示す説明図である。

〔図４Ｇ〕　図３Ｄに示す変換後矢印アドレスの内の後部アドレスの一例を示す説明図である。

〔図４Ｈ〕　図３Ｇに示す４つの変換後矢印後部アドレスから得られる軌跡生成用アドレスを示す説明図である。

【図５】　一実施の形態としての、ハンマー投げゲームの画面表示例を示す説明図である。

【図６】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。

【図７】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。

【図８】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。

【図９】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。

【図１０】ポリゴン画像表示ルーチンＳ１００による制御動作を説明するためのフローチャートである。

【図１１】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。

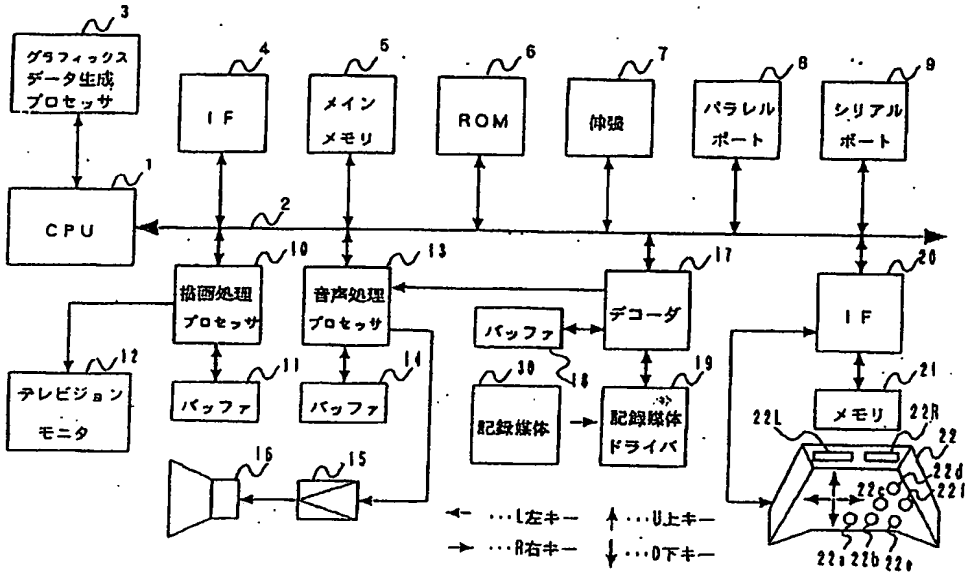
【図１２】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。

【図１３】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。

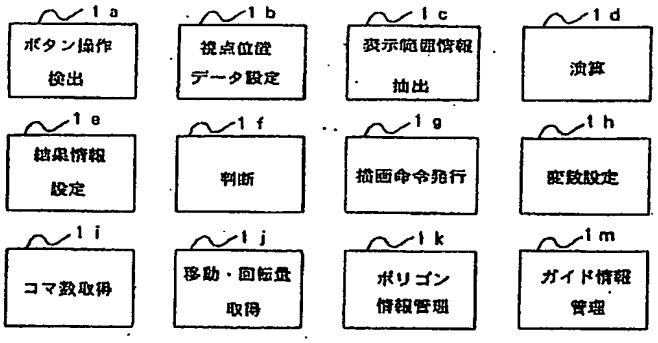
【符号の説明】

１　ＣＰＵ
１ａ　ボタン操作検出手段
１ｂ　視点位置データ設定手段
１ｃ　表示範囲情報抽出手段
１ｄ　演算手段
１ｅ　結果情報設定手段
１ｆ　判断手段
１ｇ　描画命令発行手段
１ｈ　変数設定手段
１ｉ　コマ数取得手段
１ｊ　移動・回転量取得手段
１ｋ　ポリゴン情報管理手段
１ｍ　ガイド情報管理手段
２　バス
３　グラフィックスデータ生成プロセッサ
４、２０　インターフェース回路
５　メインメモリ
６　ＲＯＭ
７　伸張回路
８　パラレルポート
９　シリアルポート
１０　描画処理プロセッサ
１１、１４、１８　バッファ
１３　音声処理プロセッサ
１５　増幅回路
１６　スピーカ
１７　デコーダ
１９　記録媒体ドライバ
２１　メモリ
２２　コントローラ

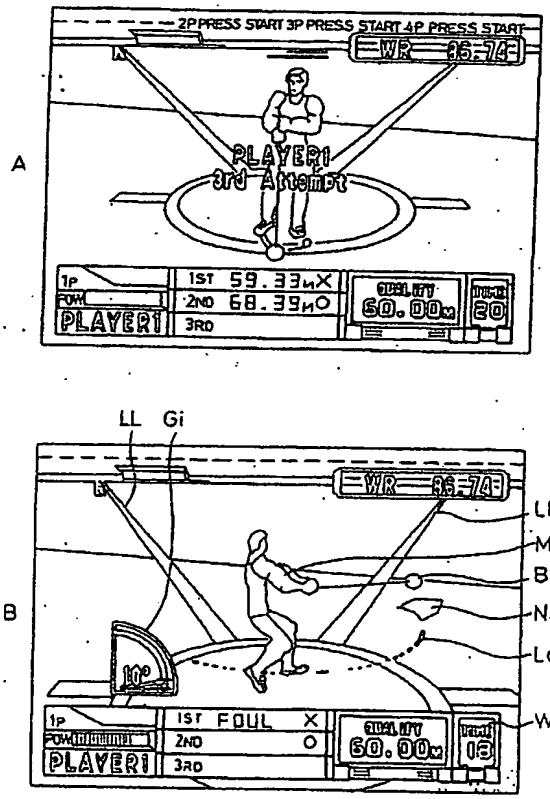
【図１】


グラフィックス
データ生成
プロセッサ
3
IF
4
メイン
メモリ
5
ROM
6
伸張
7
パラレル
ポート
8
シリアル
ポート
9
CPU
1
2
描画処理
プロセッサ
10
音声処理
プロセッサ
13
デコーダ
17
IF
20
テレビジョン
モニタ
12
バッファ
11
バッファ
14
バッファ
30
18
記録媒体
記録媒体
ドライバ
19
16
15
メモリ
21
22L
22R
22
22d
221
22c
22a 22b 22e
…L左キー
…U上キー
…R右キー
…D下キー

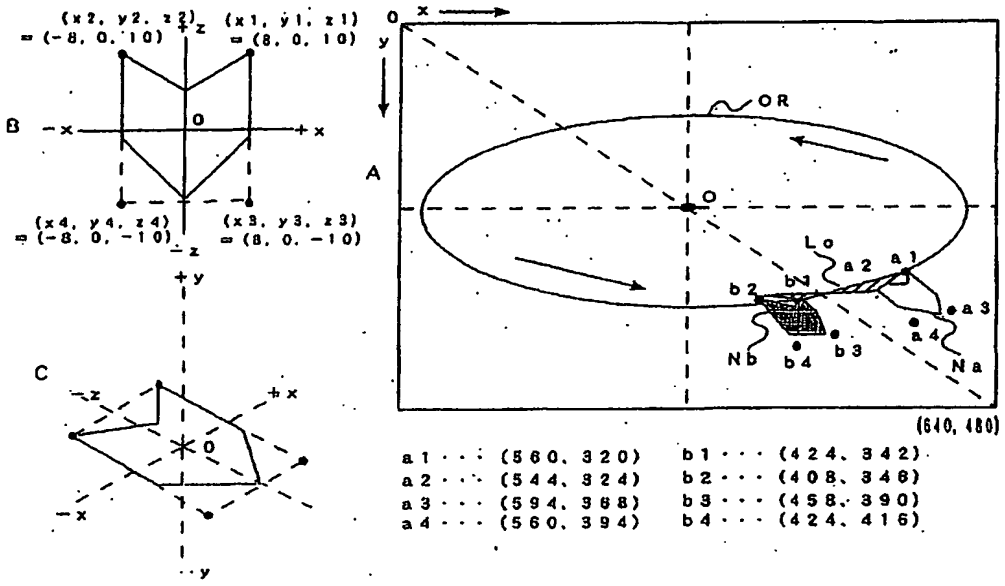
【図２】


1a
ボタン操作
検出
1b
視点位置
データ設定
1c
表示範囲情報
抽出
1d
演算
1e
結果情報
設定
1f
判断
1g
描画命令発行
1h
距離設定
1i
コマ数取得
1j
移動・回転量
取得
1k
ポリゴン
情報管理
1m
ガイド情報
管理

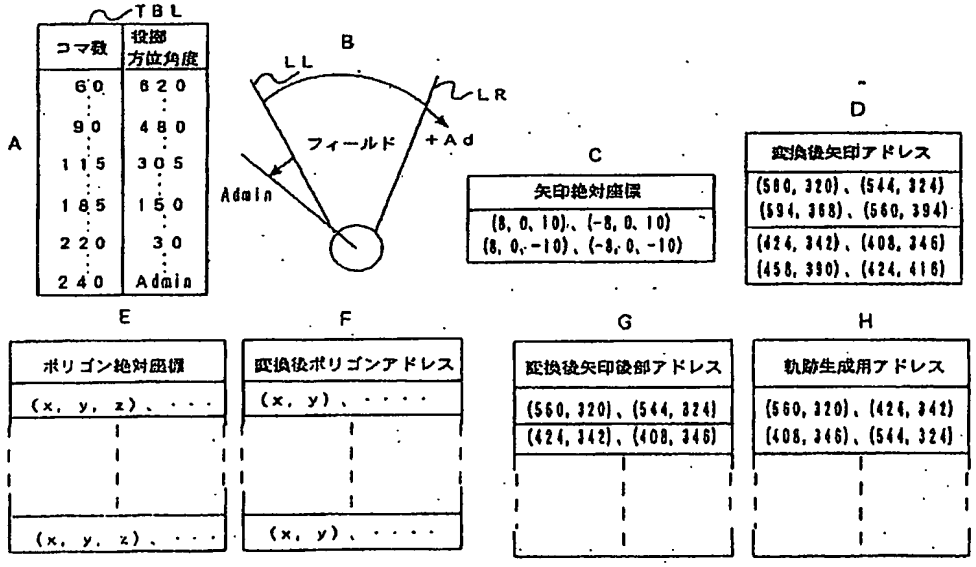
【図５】


A
2P PRESS START 3P PRESS START 4P PRESS START
WR 86.74
PLAYER1
3rd Attempt
1P
POW
PLAYER1
1ST 59.33M X
2ND 68.39M O
3RD
DISTANCE
60.00M
TRIAL
20
B
LL
Gi
WR 86.74
LR
Ma
Ba
Na
Lo
W
10°
1ST FOUL X
2ND O
3RD
DISTANCE
60.00M
TRIAL
18

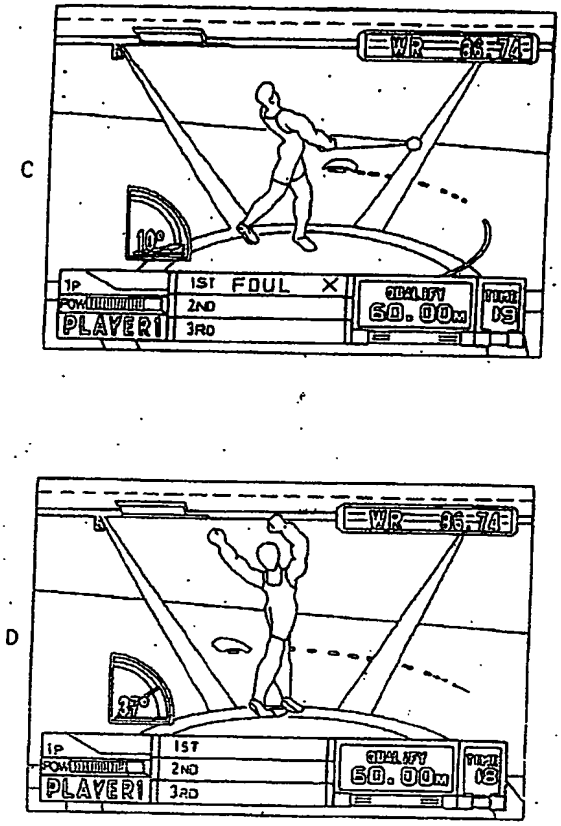
【図３】


(x2, y2, z2) = (-8, 0, 10)
(x1, y1, z1) = (8, 0, 10)
(x4, y4, z4) = (-8, 0, -10)
(x3, y3, z3) = (8, 0, -10)
B
C
A
OR
Lo
Nb
Na
(640, 480)
a1 ・・・ (560, 320)
a2 ・・・ (544, 324)
a3 ・・・ (594, 368)
a4 ・・・ (560, 394)
b1 ・・・ (424, 342)
b2 ・・・ (408, 346)
b3 ・・・ (458, 390)
b4 ・・・ (424, 416)

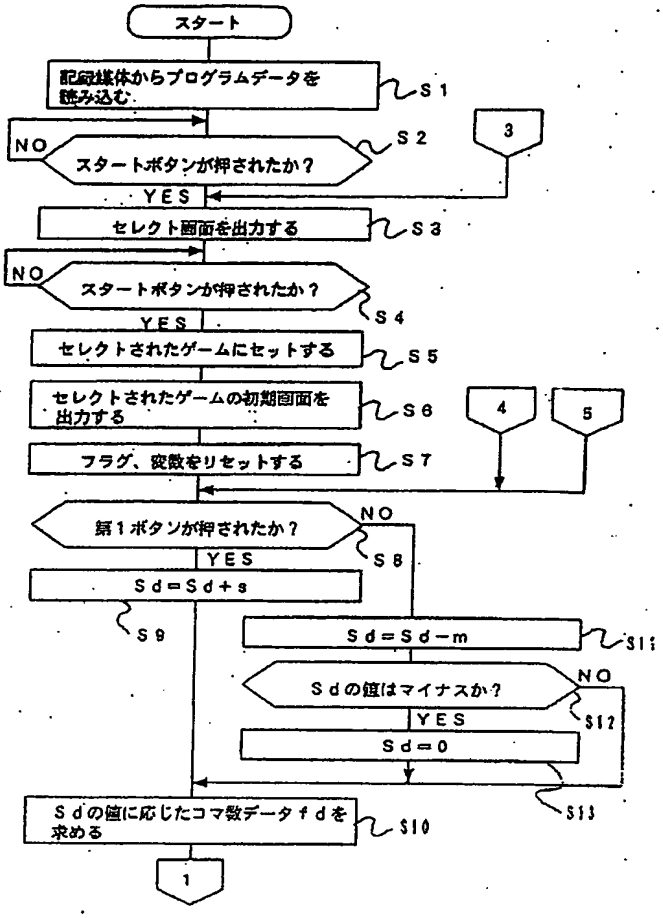
【図４】


TBL
コマ数
役廻方位角度
60 820
90 480
115 305
185 150
220 30
240 Admin
A
B
LL
LR
フィールド
+Ad
Admin
C
矢印絶対座標
(8, 0, 10)、(-8, 0, 10)
(8, 0, -10)、(-8, 0, -10)
D
変換後矢印アドレス
(560, 320)、(544, 324)
(594, 368)、(560, 394)
(424, 342)、(408, 346)
(458, 390)、(424, 416)
E
ポリゴン絶対座標
(x, y, z)、・・・
(x, y, z)、・・・
F
変換後ポリゴンアドレス
(x, y)、・・・・
(x, y)、・・・・
G
変換後矢印後部アドレス
(560, 320)、(544, 324)
(424, 342)、(408, 346)
H
軌跡生成用アドレス
(560, 320)、(424, 342)
(408, 346)、(544, 324)

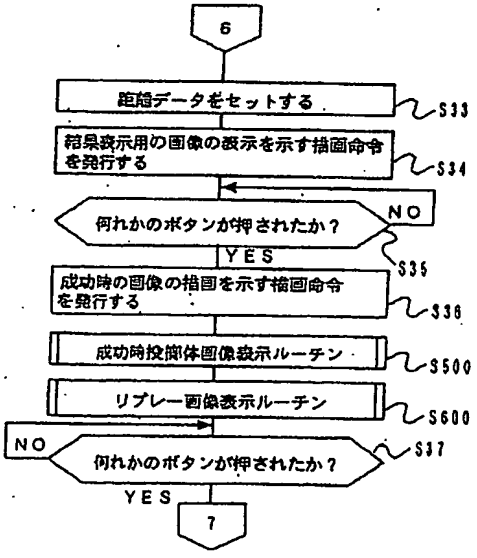
【図6】


C
WR 86.74
110°
1P
POW
PLAYER1
1ST FOUL X
2ND
3RD
QUALIFY
60.00M
TIME
D
WR 86.74
D
350°
1P
POW
PLAYER1
1ST
2ND
3RD
QUALIFY
60.00M
TIME
D

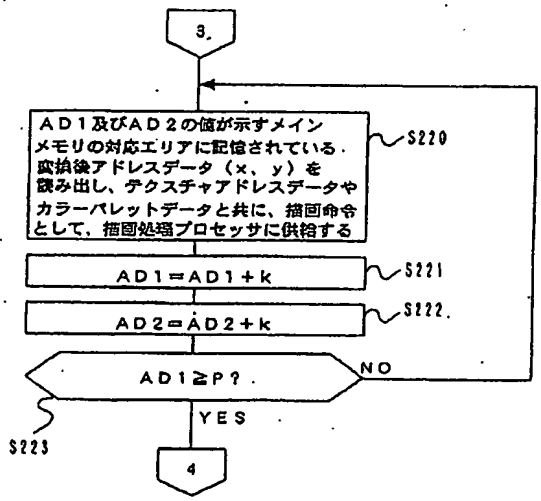
【図7】


スタート
記録媒体からプログラムデータを読み込む
S1
NO
スタートボタンが押されたか？
S2
3
YES
セレクト画面を出力する
S3
NO
スタートボタンが押されたか？
S4
YES
セレクトされたゲームにセットする
S5
セレクトされたゲームの初期画面を出力する
S6
4
5
フラグ、変数をリセットする
S7
第1ボタンが押されたか？
NO
S8
YES
Sd=Sd+s
S9
Sd=Sd-m
S11
Sdの値はマイナスか？
NO
S12
YES
Sd=0
S13
Sdの値に応じたコマ数データfdを求める
S10
1

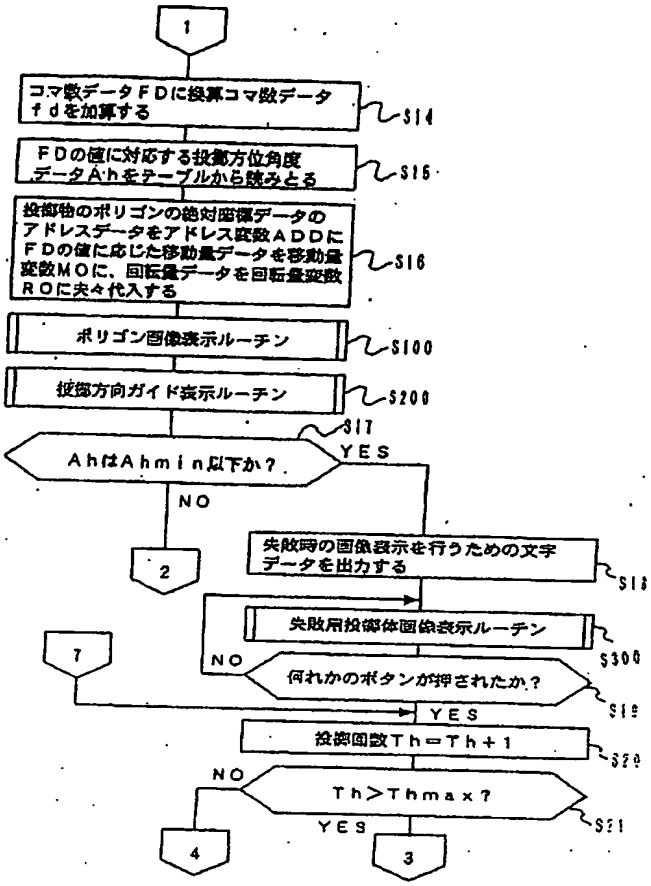
【図10】


6
距離データをセットする
S33
結果表示用の画像の表示を示す描画命令を発行する
S34
何れかのボタンが押されたか？
NO
S35
YES
成功時の画像の描画を示す描画命令を発行する
S36
成功時投擲体画像表示ルーチン
S500
リプレー画像表示ルーチン
S600
NO
何れかのボタンが押されたか？
S37
YES
7

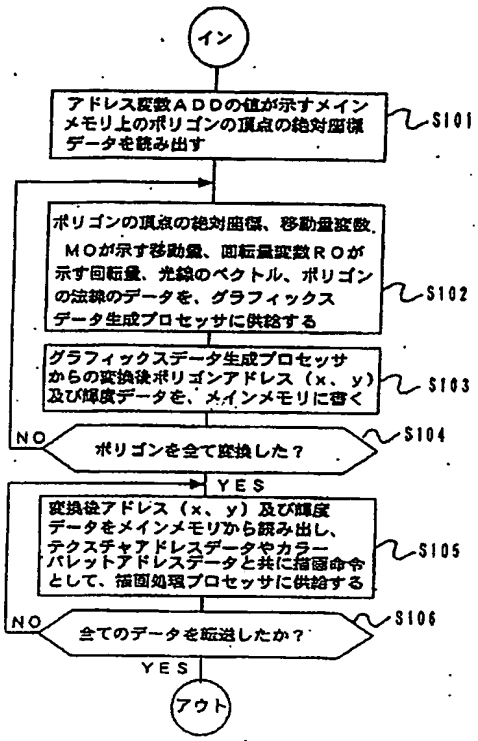
【図14】


3.
AD1及びAD2の値が示すメインメモリの対応エリアに記憶されている、変換後アドレスデータ（x、y）を読み出し、テクスチャアドレスデータやカラーパレットデータと共に、描画命令として、描画処理プロセッサに供給する
S220
AD1=AD1+k
S221
AD2=AD2+k
S222
AD1≧P？
NO
S223
YES
4

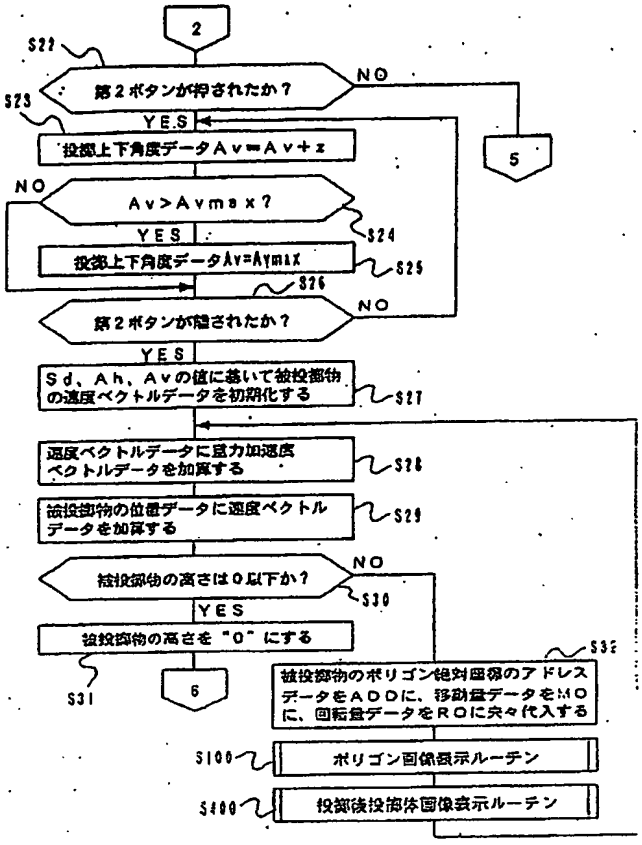
【図８】


1
コマ数データＦＤに投算コマ数データｆｄを加算する
S14
ＦＤの値に対応する投御方位角度データＡｈをテーブルから読みとる
S15
投御物のポリゴンの絶対座標データのアドレスデータをアドレス変数ＡＤＤにＦＤの値に応じた移動量データを移動量変数ＭＯに、回転量データを回転量変数ＲＯに夫々代入する
S16
ポリゴン画像表示ルーチン
S100
投御方向ガイド表示ルーチン
S200
S17
ＡｈはＡｈｍｉｎ以下か？
YES
NO
2
失敗時の画像表示を行うための文字データを出力する
S18
失敗用投御体画像表示ルーチン
S300
NO
何れかのボタンが押されたか？
S19
7
YES
投御回数Ｔｈ＝Ｔｈ＋１
S20
NO
Ｔｈ＞Ｔｈｍａｘ？
S21
YES
4
3

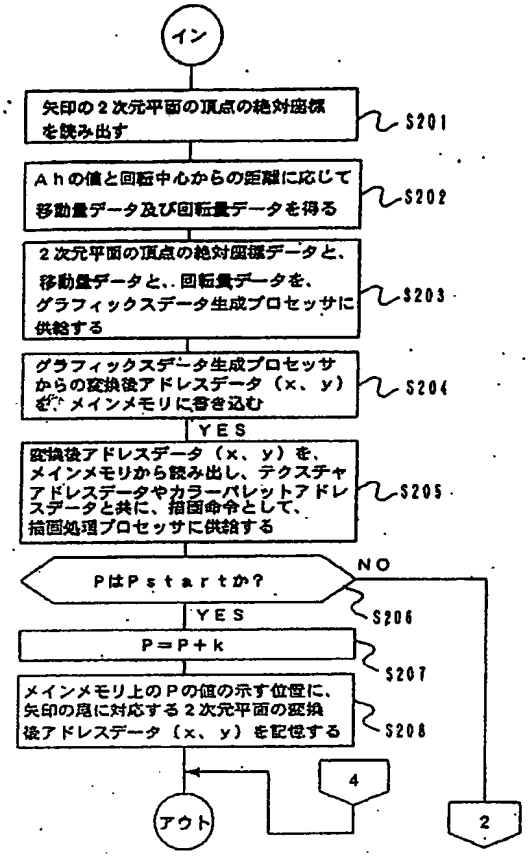
【図１１】


イン
アドレス変数ＡＤＤの値が示すメインメモリ上のポリゴンの頂点の絶対座標データを読み出す
S101
ポリゴンの頂点の絶対座標、移動量変数ＭＯが示す移動量、回転量変数ＲＯが示す回転量、光線のベクトル、ポリゴンの法線のデータを、グラフィックスデータ生成プロセッサに供給する
S102
グラフィックスデータ生成プロセッサからの変換後ポリゴンアドレス（ｘ、ｙ）及び輝度データを、メインメモリに書く
S103
NO
ポリゴンを全て変換した？
S104
YES
変換後アドレス（ｘ、ｙ）及び輝度データをメインメモリから読み出し、テクスチャアドレスデータやカラーパレットアドレスデータと共に描画命令として、描画処理プロセッサに供給する
S105
NO
全てのデータを転送したか？
S106
YES
アウト

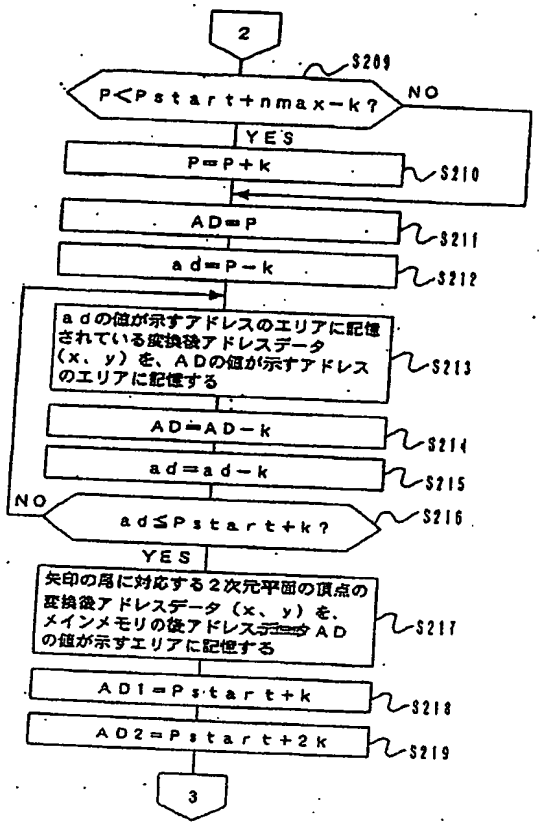
【図９】


2
S22
第２ボタンが押されたか？
NO
YES
S23
投擲上下角度データＡｖ＝Ａｖ＋ｚ
5
NO
Ａｖ＞Ａｖｍａｘ？
YES
S24
投擲上下角度データＡｖ＝Ａｖｍａｘ
S25
S26
第２ボタンが離されたか？
NO
YES
Ｓｄ、Ａｈ、Ａｖの値に基いて被投擲物の速度ベクトルデータを初期化する
S27
速度ベクトルデータに重力加速度ベクトルデータを加算する
S28
被投擲物の位置データに速度ベクトルデータを加算する
S29
被投擲物の高さは０以下か？
NO
S30
YES
被投擲物の高さを"０"にする
S31
6
S32
被投擲物のポリゴン絶対座標のアドレスデータをＡＤＤに、移動量データをＭＯに、回転量データをＲＯに夫々代入する
S100
ポリゴン画像表示ルーチン
S400
投擲後投擲体画像表示ルーチン

【図１２】


イン
矢印の２次元平面の頂点の絶対座標を読み出す
S201
Ａｈの値と回転中心からの距離に応じて移動量データ及び回転量データを得る
S202
２次元平面の頂点の絶対座標データと、移動量データと、回転量データを、グラフィックスデータ生成プロセッサに供給する
S203
グラフィックスデータ生成プロセッサからの変換後アドレスデータ（ｘ、ｙ）をメインメモリに書き込む
S204
YES
変換後アドレスデータ（ｘ、ｙ）を、メインメモリから読み出し、テクスチャアドレスデータやカラーパレットアドレスデータと共に、描画命令として、描画処理プロセッサに供給する
S205
ＰはＰｓｔａｒｔか？
NO
S206
YES
Ｐ＝Ｐ＋ｋ
S207
メインメモリ上のＰの値の示す位置に、矢印の尾に対応する２次元平面の変換後アドレスデータ（ｘ、ｙ）を記憶する
S208
4
アウト
2

【図13】



【手続補正書】
【提出日】平成9年7月17日
【手続補正1】
【補正対象書類名】明細書
【補正対象項目名】図面の簡単な説明
【補正方法】変更
【補正内容】
【図面の簡単な説明】
【図1】本発明の一実施の形態を示すゲームシステムの構成図である。
【図2】図1に示したCPU1が有する機能を示す機能ブロック図である。
【図3】矢印及び軌跡を説明するための説明図である。
【図3A】　軌道と矢印の関係及び軌跡の一例を示す説明図である。
【図3B】　矢印の絶対座標の一例を示す説明図である。
【図3C】　矢印の絶対座標から疑似3次元表示する際の概念を説明するための説明図である。
【図4】テーブル及びメインメモリ上における画像情報の状態を示す説明図である。
【図4A】　コマ数とこれに対応する矢印角度データからなるテーブルを示す説明図である。
【図4B】　投擲フィールド及び投擲方位角度データの最小値を示す説明図である。
【図4C】　矢印絶対座標の一例を示す説明図である。
【図4D】　図4Cに示す絶対座標から変換された変換後矢印アドレスの一例を示す説明図である。
【図4E】　ポリゴン絶対座標の一例を示す説明図である。
【図4F】　図4Eに示す絶対座標から変換された変換後ポリゴンアドレスの一例を示す説明図である。
【図4G】　図4Dに示す変換後矢印アドレスの内の後部アドレスの一例を示す説明図である。
【図4H】　図4Gに示す4つの変換後矢印後部アドレスから得られる軌跡生成用アドレスを示す説明図である。
【図5】一実施の形態としての、ハンマー投げゲームの

画面表示の例を示す説明図である。
【図６】一実施の形態としての、ハンマー投げゲームの画面表示の例を示す説明図である。
【図７】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。
【図８】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。
【図９】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。
【図１０】ゲームプログラムのメインルーチンによる制御動作を説明するためのフローチャートである。
【図１１】ポリゴン画像表示ルーチンＳ１００による制御動作を説明するためのフローチャートである。
【図１２】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。
【図１３】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。
【図１４】投擲ガイド表示ルーチンＳ２００による制御動作を説明するためのフローチャートである。
【符号の説明】
１　ＣＰＵ
１ａ　ボタン操作検出手段
１ｂ　視点位置データ設定手段
１ｃ　表示範囲情報抽出手段
１ｄ　演算手段
１ｅ　結果情報設定手段
１ｆ　判断手段
１ｇ　描画命令発行手段
１ｈ　変数設定手段
１ｉ　コマ数取得手段
１ｊ　移動・回転量取得手段
１ｋ　ポリゴン情報管理手段
１ｍ　ガイド情報管理手段
２　バス
３　グラフィックスデータ生成プロセッサ
４、２０　インターフェース回路
５　メインメモリ
６　ＲＯＭ
７　伸張回路
８　パラレルポート
９　シリアルポート
１０　描画処理プロセッサ
１１、１４、１８　バッファ
１３　音声処理プロセッサ
１５　増幅回路
１６　スピーカ
１７　デコーダ
１９　記録媒体ドライバ
２１　メモリ
２２　コントローラ